NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

ANALYZING THE FEASIBILITY OF USING SECURE APPLICATION INTEGRATION METHODOLOGY (SAIM) FOR INTEGRATING DON ENTERPRISE RESOURCE PLANNING (ERP) APPLICATIONS

by

Ramon O. Marin

March 2004

Thesis Advisor:	Glenn R. Cook
Second Reader:	Douglas E. Brinkley

Approved for public release; distribution unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE	Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	March 2004	Master's Thesis

4. TITLE AND SUBTITLE: Analyzing the Feasibility of Using Secure Application Integration Methodology (SAIM) for Integrating DON Enterprise Resource Planning (ERP) Applications	5. FUNDING NUMBERS
6. AUTHOR(S) Ramon O. Marin	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A	10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT	12b. DISTRIBUTION CODE
Approved for public release; distribution unlimited	

13. ABSTRACT *(maximum 200 words)*

A thorough examination of the Secure Application Integration Methodology (SAIM) for applicability in the Department of the Navy (DON) would provide useful information about a beneficial methodology. SAIM is analyzed, by accessing its step by step directions, for suitability in the integration of the Enterprise Resource Planning (ERP) projects implemented by the SYSTEMS COMMANDS (SYSCOMS).

The Navy Enterprise Convergence Team (NECT) that leads the ERP integration effort could benefit from a sound Enterprise Application Integration methodology. Results do not support SAIM as the sole guiding EAI methodology however it could have some value to the NECT.

SAIM has three primary benefits which NECT could employ: 1) It provides a complete walkthrough of the EAI process, 2) It emphasizes the importance of an Enterprise Architecture, and 3) It provides useful management checklists along with other important considerations.

SAIM also has some significant shortcomings: 1) It does not support all the DON Chief Information Officer requirements, 2) It does not provide Change Management Guidance, 3) It does not take into account the uniqueness of the Navy's environment, and finally 4) SAIM relies on an Enterprise Architecture as its foundation which the Navy does not currently have.

14. SUBJECT TERMS Enterprise Resource Planning, Secure Application Integration Methodology, ERP Convergence, Enterprise Application Integration	15. NUMBER OF PAGES 119
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution unlimited

ANALYZING THE FEASIBILITY OF USING SECURE APPLICATION INTEGRATION METHODOLOGY (SAIM) FOR INTEGRATING DON ENTERPRISE RESOURCE PLANNING (ERP) APPLICATIONS

Ramon O. Marin
Lieutenant Commander, United States Navy
B.S., United States Naval Academy, 1992

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN INFORMATION TECHNOLOGY MANAGEMENT

from the

NAVAL POSTGRADUATE SCHOOL
March 2004

Author: Ramon O. Marin

Approved by: Glenn R. Cook
Thesis Advisor

Douglas E. Brinkley
Second Reader

Dan C. Boger
Chairman, Department of Information Science

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

A thorough examination of the Secure Application Integration Methodology (SAIM) for applicability in the Department of the Navy (DON) would provide useful information about a beneficial methodology. SAIM is analyzed, by accessing its step by step directions, for suitability in the integration of the Enterprise Resource Planning (ERP) projects implemented by the SYSTEMS COMMANDS (SYSCOMS).

The Navy Enterprise Convergence Team (NECT) that leads the ERP integration effort could benefit from a sound Enterprise Application Integration methodology. Results do not support SAIM as the sole guiding EAI methodology however it could have some value to the NECT.

SAIM has three primary benefits which NECT could employ: 1) It provides a complete walkthrough of the EAI process, 2) It emphasizes the importance of an Enterprise Architecture, and 3) It provides useful management checklists along with other important considerations.

SAIM also has some significant shortcomings: 1) It does not support all the DON Chief Information Officer requirements, 2) It does not provide Change Management Guidance, 3) It does not take into account the uniqueness of the Navy’s environment, and finally 4) SAIM relies on an Enterprise Architecture as its foundation which the Navy does not currently have.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

Thank you for your guidance and support. This thesis was greatly improved by the patience and contributions provided by:

Glenn Cook, Thesis Advisor

Professor Douglas E. Brinkley, Second Reader

Marisela T. Marin

William J. Hinz

Jacob C. Hinz

Blanca E. Marin

Jose E. Marin, JR

Aurora Marin

Pam Silva

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. BACKGROUND

In 1998, the Department of the Navy's Commercial Best Practices Executive Steering Group (ESG) selected Enterprise Resource Planning (ERP) to modernize technology and business processes. This was part of an effort to revolutionize business affairs in order to offset lower budgets, mandated savings, and increased workload.

The ESG authorized four ERP pilot projects to assess Enterprise Applications. Each Systems Command (SYSCOM), including Naval Supply (NAVSUP), Naval Sea (NAVSEA), Naval Air (NAVAIR) and Space and Naval Warfare (SPAWAR), implemented an ERP pilot project. Specifically, these Enterprise Applications were to provide the following functions:

- Provide timely and rapid access to information and readiness metrics.
- Support Total Asset Visibility.
- Enhance the Planning and Scheduling Process.
- Provide Better Decision Making Tools.
- Reduce Total Cost of Ownership.
- Minimize and Simplify Data Collection.
- Utilize Common Processes across the Enterprise.

In 2002, the success of the ERP pilot projects led the Assistant Secretary of the Navy for Research, Development and Acquisition (RDA) to direct a convergence of the four projects, and the premise being that integrating the four ERP pilot projects would result in improved inter-operability, reduced total costs and an optimized Navy enterprise by:

- Focusing on the Fleet.
- Providing End-To-End product management.
- Provide greater reengineering opportunities.
- Standardized processes.

The integration of the four ERP pilot projects is an Enterprise Application Integration (EAI) endeavor. While commercial industry has aggressively pursued Enterprise Application Integration, DON just recently began focusing efforts on integrating Enterprise Application Systems and is lacking a standardized method for conducting an effective Enterprise Application Integration.

B. PURPOSE

Since Enterprise Application Integration is new to the Department of the Navy (DON), it can benefit from Enterprise Application Integration methodologies being used in private industry. Private industry uses various methodologies to conduct an effective Enterprise Application Integration. Secure Application Integration Methodology (SAIM) is one such tool used in private industry to facilitate effective Enterprise Application Integration.

SAIM is to Enterprise Application Integration as “system design is to the system itself” (Ruh, Maginnis, & Brown, 2001, p. 154). The purpose of this thesis is to determine if SAIM can be applied to mitigate risks associated with integrating the four ERP pilot projects. The intent is to apply SAIM at a high level to SYSCOMS’ ERP convergence effort.

C. RESEARCH QUESTIONS

1. Primary

Can Secure Application Integration Methodology (SAIM) be applied to the SYSCOMS' ERP convergence effort?

2. Secondary

Can using Secure Application Integration Methodology (SAIM) mitigate risk in the Navy's ERP convergence effort?

D. EXPECTED BENEFITS OF STUDY

Research will focus on the Secure Application Integration Methodology (SAIM) and will determine if such a model is suitable for use in convergence of the four ERP systems in DON. The results will be of interest to decision makers that plan to integrate ERP software across DON.

E. SCOPE OF THESIS

This thesis will rely on the collection and evaluation of data related to Enterprise Application Integration. Data will then be applied and analyzed in the integration effort of the four Navy ERP systems. More specifically, the Secure Application Integration Methodology (SAIM) will be investigated for suitability and applicability to SYSCOMS ERP convergence efforts.

F. METHODOLOGY

The goal of this research is to study, understand and interpret SAIM, which is designed to facilitate Enterprise Application Integration, in relation to the convergence of four ERP systems in DON. The basic principles of SAIM are to align IT with the enterprise business strategy, build on a solid enterprise architecture; leverage legacy and commercial software and focus on security (Ruh et al., 2001).

This research approach will rely on historical and case study research methods.

1. Historical Research Methods

Historical data will be obtained from both primary and secondary sources (Gay & Airasian, 1996).

a. Primary Sources

(1) Commercial Sources. The purpose of commercial sources will be to provide commercial "smart practices" in regards to Enterprise Application Integration. Currently, there is no single recipe for converging disparate ERP systems. This thesis will investigate current methods by interviewing personnel in the field, analyzing technical documents and sampling current commercial off the shelf (COTS) technologies that promises to assist in the Enterprise Application Integration effort.

Primary sources will include interviews with personnel actively involved in Enterprise Application Integration efforts. Such interviews will be conducted with IBM executives who are currently leading the ERP conversion effort at IBM. Interviews will also be conducted with IBM process improvement experts and IBM change management consultants.

A second primary source will include project management, change management and Enterprise Application Integration technical documents being used by private contractors throughout the department of the Navy in Enterprise Application projects.

Lastly, a third primary source will be a top level evaluation of COTS technologies designed to enable Enterprise Application Integration. Technical reports of implemented COTS tailored to facilitate the integration of

different technologies will be reviewed. Such technology includes, but is not limited to, Web-Services and Dynamic Web Application. These systems are promising enterprise level and legacy system integration.

(2) Military Sources. The purpose of military sources will be to provide an in-depth look at the military Enterprise Application environment in order to determine if it is suitable to adopt commercial industry's "smart practices."

The desirable goal here is to accomplish what Bardach notes as "looking at both the source contexts, where the practice appears to have worked well, and at your own target context, where it is being considered for adoption" (Bardach, 2000, p. 83).

The military environment will be determined by interviews, current policy and technical documents, researcher participation and researcher experience. Once the military environment has been assessed, the "smart practices" from the commercial industry, they will be analyzed to determine their applicability to the DON Enterprise Application Integration effort.

Interviews will be conducted with the Commanding Officers of different units, NAVSUP and NAVSEA stakeholders and various end users. One of the Commanding Officers will be from FISC Jacksonville detachment Ingleside who is a key member of NAVSUP organization. Another Commanding Officer will be from Shore Intermediate Maintenance Activity Ingleside and he is a key member of the NAVSEA organization.

These two units are at the forefront of Enterprise Application Integration efforts and have experience in implementing Enterprise Applications and

leading change. Commanding Officers from units will be able to provide insight into the change management perspective from a senior management point of view. NAVSUP and NAVSEA stakeholder interviews will be conducted to gather information on recent trends and thoughts on decisions regarding Enterprise Applications and change management in general.

During interviews, the researcher will be particularly interested in accounts of events, participant's interpretations of those events and the results of those events. The goal is to interview multiple participants with similar backgrounds to compare data for validity and applicability.

The researcher will review current technical and policy guidance in the undertakings of various Enterprise Applications implementations that are either in progress or have recently been completed in the Department of the Navy.

b. Secondary Sources

(1) Commercial Sources. The Historical Research Method will also rely on secondary sources for key information. Secondary sources will include professional publications such as APICS, CIO and other relevant Information Technology, Change Management and Project Management literature. This literature is important to understanding commercial industry trends and, therefore, will be relied upon to support this study.

Another secondary resource will be literature related to Systems Applications and Products in Data Processing (SAP) R/3; the backbone of each of the four SYSCOMS' ERP systems.

(2) Military Sources. Secondary sources will also be used to provide a top-level view of the elements in the military that support the Navy's ERP project. These sources will include a cursory look at the promotion and advancement system and its affects on personnel tasked to implement change. Sources will include a general study of the Navy's acquisition system and its effects on project management, budgets and schedules. This type of source will also be used to address, in general terms, the affects of the Navy's chain of command philosophy on technology and innovation. Lastly, secondary sources will be used to analyze current trends and policy guidance that may affect the ERP convergence effort.

2. Case Studies Research Methods

The Case Studies Research Method involves analyzing case studies and articles relating to Enterprise Application Integration in commercial industry. Case studies and articles will be evaluated for similarities to an Enterprise Application Integration in the Navy.

a. Commercial Case Studies

Commercial case studies of organizations that have undergone similar changes, either as a whole or in part, will provide valuable information as to what can be expected when Navy Enterprise Application Integration is implemented. This research will capture those lessons learned so that they may be applied and leveraged, if feasible, in the Navy's Enterprise Application Integration efforts.

b. Military Case Studies

Military case studies will be used to determine how personnel attitudes, processes and military culture react to change, associated with the implementation of new

technology. Information obtained from military case studies will be used to predict suitability of “smart practices” in a military environment.

G. ORGANIZATION OF STUDY

Chapter II introduces Enterprise Application Integration along with current and future trends. It also introduces the Secure Application Integration Methodology (SAIM) and provides a brief description of its elements.

Chapter III analyses the ERP pilot project implemented by the SYSCOMS and the ERP convergence efforts in DON. It commences with a description of ERP and management’s decision to implement it and proceeds to analyze, at a high level, each of the pilot programs and concludes by highlighting the convergence effort.

Chapter IV discusses Secure Application Integration Methodology (SAIM) in light of Navy’s efforts to converge the SYSCOMS ERP pilot projects.

Chapter V answers the primary and secondary questions of this thesis. It also provides recommendations and conclusions on Enterprise Application Integration in DON and identifies areas for further research.

II. ENTERPRISE APPLICATION INTEGRATION (EAI)

A. ENTERPRISE APPLICATION INTEGRATION

Enterprise Application Integration (EAI) refers to utilizing modern techniques to interconnect individual components, “each optimally designed for partial functionality,” (Gillmann, Herter, Jung, Kaufmann, & Wolber, 2002, p. 604) in order to provide an application of optimal design. EAI attempts to marry business processes and technology application needs.

EAI entails bringing together individual systems into a common enterprise system that results in centrally managed common processes based on standard operating procedures. EAI depends on each individual component’s ability to communicate from its native location to any remote application that calls it to service. Linthicum provides a short but very descriptive definition of EAI;

> EAI is able to take many diverse systems and bundle them in such a way that they appear-and function-as a monolithic and unified application (Linthicum, 2000, p. 16).

In order to ensure a successful EAI, there must be a justifiable need for integration, an integration plan, and a method to execute the integration plan.

B. THE NEED FOR ENTERPRISE APPLICATION INTEGRATION

Until recently, organizations have traditionally assembled their technology applications in a piecemeal manner in an attempt to stay current with technological trends. Often the technology added was not part of any strategic plan and was implemented without considering future interface needs. The result was often an expensive

and hard to manage EA system of incompatible applications patched together.

In the meantime, the continuous evolution of technology promises to allow varied and disparate applications to be integrated efficiently. The proper integration of enterprise applications provides a networked IT infrastructure which “dramatically reduces the cost of maintaining and running the IT infrastructure of a company or industry while simultaneously improving the functionality needed to support critical business operation” (Applegate, Austin, & McFarlan, 2003, p. 276).

1. Evolution of EA

Enterprise Applications can be categorized into four types, including Traditional Systems, Microcomputer Systems, Distributed Systems and Packaged Applications (Linthicum, 2000).

a. Traditional Systems

These applications are technically isolated and indispensable to the enterprise. These systems support essential business processes and cannot be easily “swapped-out” or replaced. Often replacing traditional systems is not feasible or is too expensive.

b. Microcomputer Systems

Commonly known as personal computers, each one is unique and acts as an enterprise application system by locally integrating both business processes and data.

c. Distributed Systems

Systems in this category cover “workstation servers and hosts tied together by a network that supports any number of applications,” (Linthicum, 2000, p. 13),

including Local Area Networks, Wide Area Networks and their associated architectures.

d. Packaged Applications

These systems are of special interest since their popularity and costs will dictate the direction of EAI. In some cases, the implementation of a packaged application solves integration issues presented by the other three application systems.

Some of the first packaged applications were in the form of Material Requirement Planning (MRP) systems, Capacity Requirements Planning (CRP) and Manufacturing Resource Planning (MRPII) which later evolved to Enterprise Resource Planning (ERP), and most recently WEB ERP, commonly known as ERPII. WEB ERP is predicted to be the dominant future packaged application.

(1) MRP. The beginnings of packaged applications can be traced back to the early 1960's when the "development and installation of computer-based MRP systems" (Plossl, 1994, p. xix) took root.

In its early design, Materials Requirements Planning (MRP) improved manufacturing processes by using new methods and technology to automate existing processes and devise new methods that resulted in a more efficient manufacturing process.

MRP automation focused on predicting supply and demand in order to properly align input with output, and therefore, minimized wasted effort and resources. This process resulted in improved production efficiency.

Since the creation of MRP, manufacturing processes have continued to evolve very rapidly and keep

fueling information technology growth as they pursue greater improved efficiency, productivity and profit.

(2) CRP. As MRP evolved and more people became familiar with its methodology, users realized that a "capacity plan" (Jakovljevic, 2000, p. 2) was needed. With the addition of capacity planning, an improved MRP emerged as Capacity Requirement Planning (CRP) which was used to plan and schedule the capacity and loading of work centers.

(3) MRPII. Packaged applications continue to evolve as Material Requirements Planning (MRP) and Capacity Requirement Planning (CRP) which were combined with "engineering, marketing, and cost data-handling programs" (Plossl, 1994, p. 204) and resulted in a broader application, Manufacturing Resource Planning (MRPII), the Roman Numeral II is used to distinguish Material Requirements Planning (MRP) from Manufacturing Resource Planning (MRPII). MRPII encompassed processes beyond the realm of manufacturing as it took into account "current available and future planned resources including capacity, space, and working capital" (Plossl, 1994, p. 205).

(4) ERP. As organizations realized the benefits of using MRPII throughout the enterprise, ERP emerged. ERP took on the role of integrating the whole organization, not just the manufacturing process. ERP functions include:

- Integrates all company's components by using a multi-module application software.
- Allows the sharing of data amongst all company components via data transmission lines.
- Uses software to automate processes, and in some cases, improves business process.

- A successful implementation results in one software application package delivering service to the entire enterprise using a single database.

Figure 1 below illustrates an ERP package with extensions, which provides the ability to share data with other programs, linking to external systems. It is important to note that within the internal organization, in this case the area labeled ERP operations are seamless, but boundaries still exist between the organization and external systems.

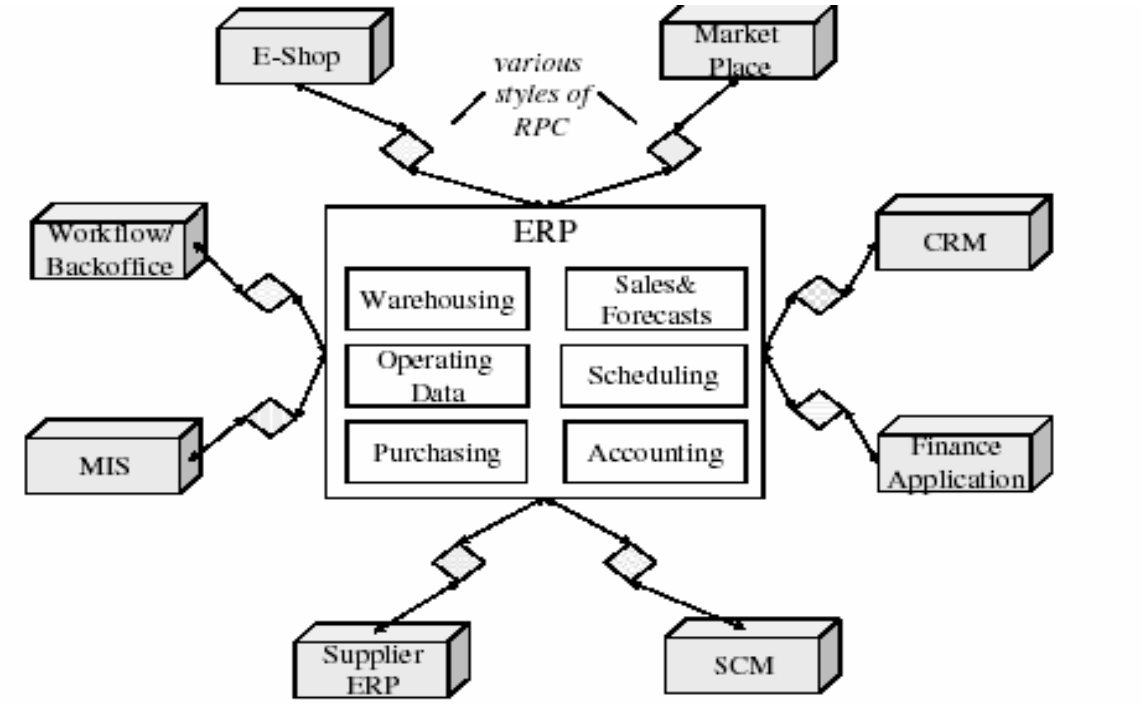

Figure 1. Monolithic ERP System with Extensions (Gillmann et al., 2002, p. 603)

As organizations master their internal processes and the market place globalizes, the need for seamless connectivity beyond internal borders of the organization continues to increase. Connectivity beyond the borders of an enterprise has been encouraged by technologies such as Application Service Providers, the Internet, and the Web, all which are part of the backbone

of the currently emerging packaged enterprise application, Web-Enabled ERP (ERPII).

(5) Web-Enabled ERP (ERPII). The Internet and globalization have created new opportunities for organizations. Organizations are now relying on the ability to access resources from outside of their information domain to maintain a competitive edge. They must be able to electronically link to their suppliers, competitors, customers and other external actors. They must also be able to receive, manipulate, update and store real-time data in order to maintain pace with current trends.

Although similar limited capability has been available in the past in the form of Electronic Data Interchange (EDI), such technology is not feasible for wide market usage.

However, the wide availability and usage of the Internet and its “cheap” infrastructure makes it a suitable and logical instrument for data interconnectivity amongst stand-alone systems in a geographically dispersed business environment.

Figure 2 illustrates an ERPII hybrid package that links to external systems. In this case, ERPII is also linking to an ERP system. It is important to note that the goal is to eliminate all boundaries between an organization and external entities.

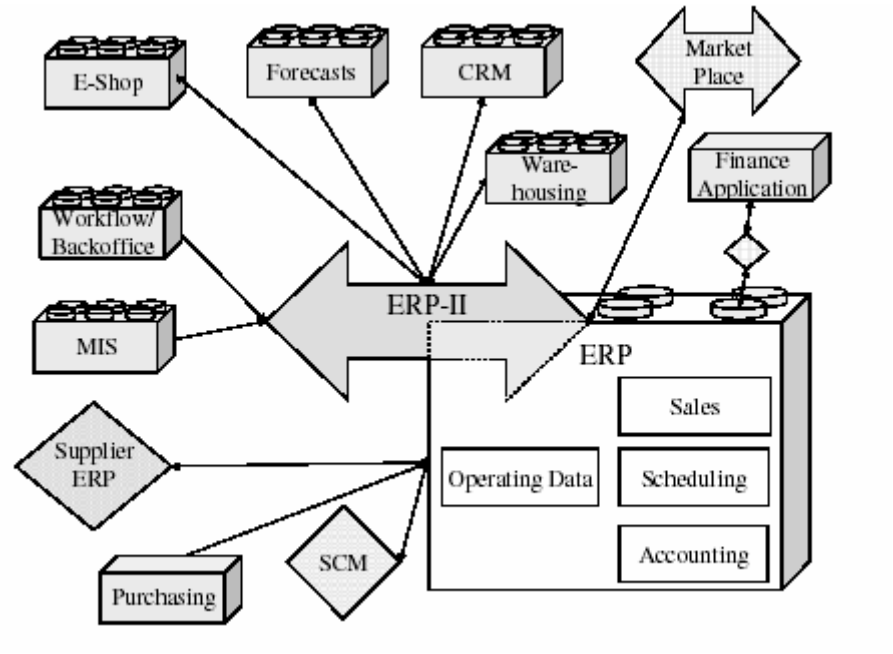

Figure 2. Coexistence of ERP and ERP-II in a Hybrid System (Gillmann et al., 2002, p. 613)

e. Emerging Technologies that Support Web-Enabled ERP (ERPII)

Currently, significant research on the technical part of "connecting" ERP exists in order for it to become Web-Enabled. As can be imagined with any emerging technology, Web-Enabled ERP is evolving as the search for standards and core technologies to support it continues. Two important technology developments are of special interest: Application Service Providers (ASP) and Web Services.

(1) Application Service Providers. Application Service Providers are a recent innovation and are at the forefront of the technology development spectrum. Application Service Providers provide the ability to run software over the Internet. This allows software to be run without physically having the software on the local computer, called a client (Gralla, 2002). Although various ways of distributing the application

exist, one thing is for certain, ASPs control the applications while organizations only have client privileges, such as viewing but not storing data locally.

(2) Web Services. Web Services are another leading emergent technology that is in a position to break the boundaries that have captured organizations and kept them within their internal informational domains. Web Services and the power of Extensible Markup Language (XML) "offers the dual promise of simplicity and pervasiveness" (Malaika, Nelin, Qu, Reinwald, & Wolfson, 2002, p. 666). In 2002, Malaika et al., summarize Web Services as;

> Web services provide a ubiquitous model for offering business services over the Internet as well as within organizations. Web services are of particular interest for their ability to incorporate third-party applications or legacy applications. In the most primitive sense, Web services can be viewed as any mechanism by which an application service may be provided to other applications on the internet (p. 666).

Web Services can be defined further as either informational or transactional. Informational Web Services provide "one way" data such as news and music. Transactional Web Services provide "two-way" data that is integrated into business-to-business infrastructure and such data may include updating records to dispersed databases. An example of Transactional Web Services is a scenario where a customer is remotely updating its supplier databases.

(3) Web Services as a Product of Technologies. Web Services are an accumulation of various technologies that have evolved in the Information Technology field. The three most important technology

tools used for Web Services are application programming interfaces (APIs), Web browsers and the Web.

One of the simple concepts used in Web Services is “to discretely package business applications by using APIs and make them accessible via the Internet to external databases and other applications” (NDS Systems, 2003).

Another tool that enables Web Services is the Web browser, where “users simply point a browser at the Web server application API to access these intuitive Web Services, and conduct on demand, automated, real time, self service transactions” (NDS Systems, 2003, para 3).

A third and the most important tool used in Web Service is the Web. The infrastructure of the Internet and the access it provides via the Web is a critical resource that Web Services is attempting to utilize fully.

(4) Core Technologies of Web Services. The emergence of Web Services has mandated new protocols and standards to support them. Malaika et al., (2002) in the article “DB2 and Web Services” describe part of the administrative effort surrounding Web Services;

> Web services are described in WSDL (Web Services Description Language). The WSDL description may be registered in the UDDI (Universal Description, Discovery, and Integration) repository. UDDI provides a set of application programming interfaces (APIs) to register and search for Web Services (p. 666).

Other important core technology supporting Web Services include SOAP (Simple Object Access Protocol) and XML. Malaika et al., (2002) in “DB2 and Web services” describe SOAP as;

> …a lightweight protocol that provides a service-oriented architecture for applications on the web. Clients compose requests and send SOAP envelopes to providers, who reply through SOAP responses (p. 666).

XML is the format of messages that are transmitted between web service applications. SOAP is XML transmitted over HTTP/SMTP (Gillmann et al., 2002, p. 610).

(5) Web Services and ROI. Many organizations are cautious after their expensive and often negative experience with an ERP implementation. They will be sure to ask, what makes Web Enabled ERP different? The answer is Web Services. Web Services promise to deliver real time information anywhere possible by providing on demand interfaces. A description of benefits expected from Web Services follows:

> …Real time automation directly translates to efficiency and increased margins, but of more interest in today's economy is competitive advantage and market share. Source Integrators have the ability to carve out and deploy intuitive "On Demand Self Service" applications from any transaction point within the enterprise. Users can initiate, query, validate, and report on transactions from any browser, anywhere, anyplace, and any time. Once originated, Web service applications can be individualistically replicated and deployed to expand the ecosystem. As a result, the focus of traditional "value statements" and "CRM" initiatives shift dramatically from "price", "availability", and "personal service" to defining new business rules and initiatives whereby the ecosystem can increase real time automated interoperability…In short Web Services "facilitate the automation of a multitude of ongoing business activities between manufacturers, distributor, dealers, and customers, (the ecosystem) (NDS Systems, 2003, para 4).

2. Impact of EAI on the Enterprise

Enterprise Application Integration (EAI) is much more than a technological solution. EAI impacts the overall success of the enterprise by weaving itself into business processes, organizational structures and organizational policy. Implementations of ERP have demonstrated that applying technology alone does not guarantee success. When ERP packages were implemented, some companies achieved significant benefits, while others failed miserably. Those that succeeded were the ones that augmented the technology with business, organizational and policy initiatives (Worthen, 2002).

a. EAI and Business Processes

EAI should not only be concerned with changes to technology infrastructure. EAI must also be concerned with its effects on the business processes. EAI systems and business processes cannot be isolated from each other. A change in one will invariably affect the other. IT and core business teams should be "working toward a common goal" (Ramankutty, 2003, p. 38). Initiatives that only focus on the technology side of change ignore "the significant benefits that can be realized from process, policy, organizational, and other types of change" (Boyle, 2003, p. 39).

Before any process can change, it must be understood in its current state. Each organization typically assembles a cross functional team to identify tasks and create a map of the current state. Once the current state is identified and all team members agree on its accurate reflection of day-to-day operations, this provides a greater chance that there is a basic

understanding of the "As-Is" state throughout the organization. Below is a summary of what Ron Crabtree in his promotion of Value Stream Mapping (VSM) suggests:

- Use paper form and send team members out to collect all the information.
- Use visual flow to discuss processes.
- Use skilled facilitator(s).
- Do not focus on fixing, but on understanding current state (Crabtree, 2003).

The "To-Be" state must be defined from within the organization. It should begin by gathering "all ideas of what can be improved" (Crabtree, 2003, p. 22) and then prioritize "the ideas based on how they affect value stream performance" (Crabtree, 2003, p. 22). This does not mean that the change being executed was not externally generated. It means that once the decision to change has been made, user involvement is critical. Users will define the requirements. Accurate requirements are essential to a successful project, as author Frame notes "…they are the embodiment of the customer's needs" and "are important because they define the project team's obligation to the customers" (Frame, 1995, p. 135).

b. EAI and Organizational Structures

Peter M. Senge believes that organizations break down "because they are unable to pull their diverse functions and talents into a productive whole" (Senge, 1990, p. 69). Information Technology serves as a catalyst for information flow but it does not break down barriers between functional entities that constitute the whole enterprise. Organizations must promote interoperability and integrate functional areas into seamless processes in

order to maximize the benefits of an Enterprise Application System.

3. Benefits of Enterprise Application Integration

EAI provides several benefits. First, it provides improved infrastructure “that streamlines highly leveraged, resource-intensive processes while layering important components of reusable infrastructure to produce measurable results in short periods of time” (Applegate et al., 2003, p. 279).

Secondly, EAI allows efficient data interchange between many sources and many destinations (Cummins, 2002). It would be very inefficient to have individual connections between each source and each destination.

Third, EAI allows data needed for transactions “to be immediately available no matter where the information is located in the enterprise” (Linthicum, 2000, p. 16).

In short, EAI packages all the diverse applications systems in the enterprise into a single system that acts as one system providing the benefits of a single enterprise application.

C. PLANNING FOR EFFECTIVE EAI

The great number of varied and disparate enterprise applications, the complexity of EAI and the lack of an EAI experience coupled with “software complexity, competitive architectures and performance issues” (Ruh et al., 2001, p. 12) mandates a disciplined and systematic approach to ensure that Enterprise Applications are successfully integrated.

In the past, applications were integrated at connection points at both the hardware and software levels. Now, integration is predominantly done at the software

level, which has increased “both the range and complexity of integration options” (Ruh et al., 2001, p. 18).

1. Types of Integration

Three integration models that minimize time, costs, and “increase the reusability and flexibility of integration” (Ruh et al., 2001, p. 19) have emerged. Models names are based on the level where applications are integrated. The three models include the presentation integration model, the data integration model and the functional integration model. A fourth model, Method level EAI that connects at the business process level has not fully matured but is worth discussing.

a. Presentation Integration Model

This model relies on the ability of graphical user interfaces (GUIs) to integrate various applications simultaneous while adding “business logic related to the management of the interface, such as validation, error checking, and calculation” (Ruh et at., 2001, p. 22). Although this is not the preferred model, it is ideal when users require a single interface for various applications or when the presentation level is the only point of integration available (Linthicum, 2000).

This type of integration is the quickest and easiest to accomplish, but since integration occurs at the user level, it is also the most limiting because “only the data and interactions defined in the legacy presentations can be accessed” (Ruh et al., 2001, p. 23). Therefore, this model does not allow access to the business logic or data processes of the enterprise. A common tool use for this type of integration is screen scraping which permits

using new technology to develop a graphical user interface from accessed legacy presentation.

b. Data Integration Model

The Data Integration Model integrates at the data level of an application and intertwines “databases or data structures of an application” (Ruh et al., 2001, p. 24) in order to integrate systems into an enterprise.

Integrating at the data level is accomplished when data from different databases is fused together to provide the information required. It is also used when data from a single source is used to feed multiple sources. Data integration is also required when the desire is to store data coming from diverse and geographically dispersed systems centrally. This prevents duplication of data and ensures data integrity by synchronizing shared data.

The benefit of integrating at the data level is that it provides greater flexibility than the presentation model, since it allows access to data at greater granularity. This greater flexibility provides data reusability and salvages legacy data, thus reducing time and effort in implementing an integrated solution. This integration approach is also cheaper because minimal re-code is needed and enabling technology is relatively inexpensive (Linthicum, 2000).

The Data Integration Model uses various tools to interconnect data from various databases and/or from different structures. Such tools include batch file transfer, Open Database Connectivity (ODBD), database access middleware, and data transformation.

c. Functional Integration Model

This model integrates data at the business logic level. Business logic refers to “implementation of business processing in a programming language” (Ruh et al., 2001, p. 27). Integration logic must be in the code of the application either as an application programming interface (API) or as code designed to allowed integration.

The most common method to facilitate integration at this level is by using distributed processing middleware;

> Distributed processing middleware is a type of software that facilitates the communication of requests between software components through the use of defined interfaces or messages...in addition it provides the runtime environment to manage the requests between software components (Ruh et at., 2001, p. 28).

The three categories of distributed processing middleware include Message Oriented Middleware (MOM,) which handles messages to and from applications; distributed object technology, which gives software the properties and functionality of objects making software accessible by other applications, and transaction processing monitors (TPMs), which support distributed architectures by managing transactions (Ruh et al., 2001, p. 28).

Functional Integration Model can be applied in three different ways, including data consistency integration, multistep process integration, and plug-and-play component integration.

Data consistency integration is useful when the requirement exists to update different records executing different business logic but dependent on the same data.

Data consistency integration is also useful when data propagation to various applications and databases is required. In this case, a command to modify, add or delete data can be sent to each of the applications or databases, in an effort to achieve data consistency throughout the enterprise.

Multistep process integration automates business processes across different applications. This integration uses steps in business processes to prioritize system tasks. Implementing this type of integration is complex since requests for actions and data can be very complicated and require that each application has the ability to maintain a current state of the request. A proper state of the request is a must to ensure proper communication amongst applications.

Plug-and-play component integration relies on the ability to add or replace components in a system without modification to the software. This type of integration relies on components adhering to common standards and interfaces with software, hardware, and business processes. Although this type of integration is the hardest to accomplish, it promises the greatest value.

d. Method Integration Model

This type of integration involves various application sharing methods. As an example, “the method for updating a customer record may be accessed from any number of applications” (Linthicum, 2000, p. 19), allows applications to share each other’s methods, thus elimination having to write each method in each application.

There are various ways of accomplishing method sharing, the most common include distributed objects and application servers.

All four types of integration are unique and each enterprise will have to evaluate their applicability. However, for best results, a combination of any or all of the integration methods may be the most suited approach. Table 1, following, provides a summary of the four methods.

Integration type	Integration Point	Ideal When	Tools
Presentation Level	Presentation	User requires single interface Integration can only be accomplished at this level	Screen Scraping
Data Level	Database or Data Structures	Data from different databases needs to be fused together Data needs to be centrally stored	Batch files transfer, Open Database Connectivity (ODBD), Database access middleware, and Data transformation
Functional Level	Business Logic	Consistency of Data across varied applications required	Distributed Processing Middleware
Method Level	Method sharing	Applications must share each other's methods	Distributed Objects and Application Servers

Table 1. Summary of Types of Integration

2. Building Blocks of the Enterprise Application Integration Architecture

Successful enterprise integration relies on a technology architecture that effectively combines models of communications, integration enabling methodology, middleware, and services to develop a physical product that can support desired functions and features expected from integrating applications.

a. Communications Models

In order for systems to interact properly, they must communicate efficiently. Communications can be categorized into two types; synchronous and asynchronous.

> Synchronous communication occurs when the communication between a sender and a receiver is accomplished in a coordinated manner. This requires sender and receiver to operate dependent on the processing of request (Ruh et al., 2001, p. 41).

The voice telephone switch is an example that uses synchronous transmission where the source and the destination must coordinate for proper communication. One sends, the other receives.

> Asynchronous Communication occurs when the communication between a sender and receiver is accomplished in a manner that allows each of them to operate independently of the other. The receiver of the request is under no obligation to handle the communications or respond to the sender. The sender continues to operate once the request is sent without regard to how the receiver handles the communication (Ruh et al., 2001, p. 45).

Electronic mail (email) communications are an example of asynchronous transmission. In this case, the sender and the receiver do not coordinate for data interchange. The sender transmits data, often at irregular intervals, without regard to the status of the receiver.

b. Integration Enabling Models

Messaging and interface definitions are two principle ways of sending and receiving data to physically integrated applications. Messaging between senders and receivers include information and data needed to accomplished the desired tasks. Interface integration relies on a “well-defined interface that describes the actions that an application can perform” (Ruh et al., 2001, p. 50).

c. Middleware Models

The middleware model is the principle facilitator of EAI. Middleware uses interfaces or messages to integrate applications. It is “a simple mechanism to move information and share business logic between applications” (Linthicum, 2000, p. 20). The middleware model makes integration easy by handling the details of complex transactions between applications. The advancement of middleware means greater ability to bridge even more varied applications into a virtual enterprise. The selection and design of middleware model is dependent on the communication and integration model for effective EAI.

d. Services Models

The services model is a critical addition to the basic technology architecture created by the communication, integration and middleware model. The service model specifies what services, not part of the core technology, will be implemented in the EAI solution. A service is defined as a “functional extension to basic communication or middleware capability” (Ruh et al., 2001, p. 57). Services facilitate development and provide needed functionality such as;

- Directory
- Lifecycle
- Security
- Conversion and transformation
- Persistence
- Events
- Notification
- Workflow (Ruh et al., 2001, p. 58)

The communications model, integration enabling model, middleware model and services model are the critical building blocks of EAI because they represent the systems that constitute and hide the complexities of the underlying technology, which implements the designed plan.

D. METHODOLOGY FOR CONDUCTING EFFECTIVE EAI

Effective EAI requires a disciplined approach. EAI has all the elements of a complex project, and therefore, must be treated as such. An effective EAI methodology provides a blueprint that must be used as a guide and strictly followed. Although this will never guarantee success, it will serve to mitigate risk. Concept Five Technologies, a company that provides EAI services, developed the Secure Application Integration Methodology (SAIM) to serve such a purpose.

> A methodology defines a coordinated set of activities that are applicable to solving problems in a specified domain. SAIM describes activities related to the effective use of EAI(Ruh et al., 2001, p. 155).

An EAI methodology should address the following:

- Ensure that the enterprise IT architecture and developed applications satisfied business needs.
- Describe how to manage the EAI process.
- Describe how to work with legacy systems and packaged solutions to integrate them.
- Provide guidance on technology selection and standardization.
- Ensures that the methodology promotes reuse (Ruh et al., 2001, p. 155).

1. Principles of SAIM

SAIM was designed to meet the fundamental requirements needed for a successful EAI. SAIM is driven by four

principles, which include alignment of IT with a business strategy, designing solid enterprise architecture, maximized benefits of legacy and new technology, and ensuring acceptable security.

2. SAIM Activity Areas

SAIM focuses EAI methodology on five areas to ensure an effective project. The first area is to ensure that Enterprise IT Strategy supports a sound EAI. The second area is to ensure that a proper Enterprise Architecture is put in place, the third focuses on Application Architecture to ensure proper hardware and software is utilized, while the fourth provides guidance on Component Development, and lastly it ensures forethought to complete Application Integration and Development.

a. Enterprise IT Strategy

The purpose of the Enterprise IT Strategy is to support the Enterprise Business Strategy. Therefore, a solid relationship must exist between the two. In this area, SAIM strives to identify strategic IT initiatives and assesses readiness for EAI.

b. Enterprise Architecture

In this area, SAIM ensures that the architect properly considers security policy, business component requirements, infrastructure requirements, legacy and packaged applications in specifying the Enterprise IT Architecture.

c. Application Architecture

This activity focuses on ensuring that the end result of EAI is a single application that will be “a software entity that focuses on providing a cohesive set of capabilities to end user[s]” (Ruh et al., 2001, p. 169).

d. Component Development

SAIM's interest in component development is at a very high level and relates to components being identified as "a software entity that provides a cohesive set of functional capabilities through a specified interface" (Ruh et al., 2001, p. 174). Example of such components includes functional systems such as a Human Resources Information System. SAIM focuses on five varied components types which include custom components, wrapped legacy application components, wrapped package applications, wrapped databases, and infrastructure components.

e. Application Integration and Development

This is the final activity area of SAIM. Its focus is to continuously monitor and improve the integration and its development effort. It relies on evaluating a pilot and performing security assessment tests.

3. Risk Management with Unprecedented Technology

Projects that involve EAI will inherently be risky. SAIM mitigates the risk by outlining certain strategies that should be followed. SAIM is particularly concerned with risk associated with the following unprecedented, yet to be evaluated, elements:

- New classes of applications
- New business domain components
- New infrastructure services
- Enterprise architecture guidelines (Ruh et al., 2001, p. 177)

A SAIM strategy to mitigate risk involves tracking high-risk situations by providing special management focus from the start. High-risk situations include:

- Business applications with extremely tight or near-term time-to-market constraints that depend on unprecedented elements.
- Mission-critical applications that depend significantly on unprecedented technology.
- Any applications that depend entirely on unprecedented elements.
- Elements that are unprecedented, not only within the enterprise, but in the marketplace as a whole (Ruh et al., 2001, p. 177).

4. Organizational Considerations

SAIM impacts the organization in two major ways. One is that it mandates an Enterprise Architecture Organization that can monitor, direct and focuses efforts toward a sound, enterprise architecture. The other is that it depends on an Information Security Steering Committee in charge of managing all security issues with authority to enforce their recommendations.

a. Enterprise Architecture Organization

The purpose of an Enterprise Architecture Organization is to ensure that within the enterprise the focus on architectural design of EAI. Architecture Organization includes:

- An Enterprise Architect, who reports to the CIO.
- An Enterprise Architecture Steering Committee, which is responsible for setting architecture policy and making major architecture decisions.
- A small Enterprise Architecture staff, who are responsible for maintaining the enterprise architecture specification, for analyzing future requirements and associated architectural changes, and for evaluating new technologies (Ruh et al., 2001, p. 178).

b. Information Security Steering Committee

Implementing a security policy in an EAI project requires continuous management that can be assisted by the following activities:

- Reviewing the enterprise's security policy in light of emergent threats.
- Investigating any breaches of security that may occur within the enterprise, and adopting rules to ensure that they are not repeated.
- Reviewing security characteristics of critical applications, both at the stage of requirements definition and before the applications are placed in production.
- Sponsoring the development of security education and training programs for the enterprise (Ruh et al., 2001, p. 178).

E. CONCLUSION

EAI is taking Enterprise Applications to the next level. In order to accomplish an EAI project successfully, a genuine need to integrate applications must exist, there must be an effective EAI plan as well as a methodology to follow. SAIM is an EAI methodology that serves as a guideline.

Chapter III will address ERP systems in the Department of the Navy (DON). It presents an overview of ERP and its integration into DON. It then discusses ERP pilot projects being conducted at the Systems Commands (SYSCOM). It concludes with a summary of the Navy Enterprise Convergence Team's (NECT) efforts to integrate ERP pilot projects.

THIS PAGE INTENTIONALLY LEFT BLANK

III. ENTERPRISE APPLICATIONS IN SYSCOMS

A. BACKGROUND

Chapter II discussed the evolution of enterprise application technology. In 1998, the enterprise application of choice was Enterprise Resource Planning (ERP).

In the 1990's, a software integrated product designed for large companies was developed by Systems Applications and Products in Data Processing (SAP), a German company. This type of product became known as an ERP system.

The idea of the ERP system offered immediate advantages over the traditional stand-alone and stove-piped systems. ERP offered most of its benefits by integrating business processes into a single system.

A single system meant that, for the first time, a dispersed organization could share the same database, same operating system and use the same business processes. P. J. Jakovljevic in his article, "Essentials of ERP-Its Functional Scope," specifically lists three reasons for implementing an ERP system; first for financial data integration, second to standardize processes, and third to standardize human resources information (Jakovljevic, 2000).

The opportunities ERP offered created a high demand for ERP systems. As implementations of ERP systems progressed, they proved to be expensive, difficult to implement and failed between "30%-50%" of the time (Cook, 2003, Slide 20). However, where implementations were successful, ERP provided a significant payoff, as an

example, “IBM reduced its time to ship a replacement part from 22 days to 3 days” (Devaraj and Kohli, 2002, p. 13).

SAP, the largest ERP Software vendor, “soared from less then $500 million in 1992 to approximately $3.3 billion in 1997” (Davenport, 2002, p. 160) thus becoming the fastest growing software company in the world. Its main competitors Oracle, PeopleSoft, JDEdwards and Baan also saw high demand for their product.

SAP continues as the market leader with approximately 32% of the market share (Cook, 2003, Slide 8). Also, over 50% of its implementations have been conducted in organizations with annual gross revenues in excess of $500 million dollars (SoftSelect Systems, 2003, p. 62).

In 1998, in response to Joint Vision 2010 and Department of the Navy (DON) Revolution in Business Affairs (RBA) goals, the Commercial Best Practices Working Group (CBPWG) was tasked with:

- Consolidating and prioritizing financial initiatives and to serve as the foundation for future reform.
- Accelerating the introduction of commercial best practices.
- Developing a strategic plan for implementing a business management process to better assess costs and performance.
- Establish a plan and architecture to implement reform.

The CBPWG originally focused on implementing commercial best *financial* practices. However, under the leadership of VADM John Lockard from Naval Air Systems Command, the reform expanded to implement commercial best *business* practices. Based on CBPWG findings, six ERP pilot

programs were authorized. As a result of reduced government funding only four of the pilots are still active, these include:

- Supply Maintenance Aviation Reengineering Team (SMART) Enterprise Resource Planning (ERP) pilot program being conducted jointly by Naval Supply Systems Command (NAVSUP) and Naval Air Systems Command (NAVAIR).
- Navy Enterprise Maintenance Automated Information System (NEMAIS) Enterprise Resource Planning (ERP Pilot program being conducted by Naval Sea Systems Command (NAVSEA).
- SIGMA Enterprise Resource Planning (ERP) Pilot program being conducted by NAVAIR.
- CABRILLO Enterprise Resource Planning (ERP) Pilot program being conducted by Space and Naval Warfare Systems Command (SPAWAR).

ERP pilot programs were chartered to assess capabilities for improving financial and general business processes in DON. All the pilot programs selected SAP as the backbone of their ERP system.

B. SYSTEMS, APPLICATIONS AND PRODUCTS IN DATA PROCESSING

SAP is both the name of the company and the software. SAP consists of integrated modules that when combined, represent most commercial business processes. SAP has coupled Information Technology with business processes in an effort to improve the efficiency of commercial and public companies. According to SAP documentation, the SAP solution is made up of standard applications, has been built on key basic principles, offers future development opportunities and is secure and reliable.

1. SAP Solution

SAP has invested considerable effort trying to ensure an efficient and successful system implementation. It offers standard business applications already configured, but available for customization. The objectives of SAP are:

- Provide a complete infrastructure for corporate information processing.
- Maintain a comprehensive repertoire of standard business functions that can be combined to model a wide range of business processes.
- Ensure that all SAP systems are usable worldwide.
- Retain a thoroughgoing open policy with respect to data access and functionality.
- Support distributed applications and interface to non-SAP systems (ASAP World Consultancy, 1999, p. 64).

The SAP software is made up of ABAP/4 Data Dictionary, BASIS System and Main SAP R/3 Applications and Components. Figure 3 is a depiction of a typical SAP Solution.

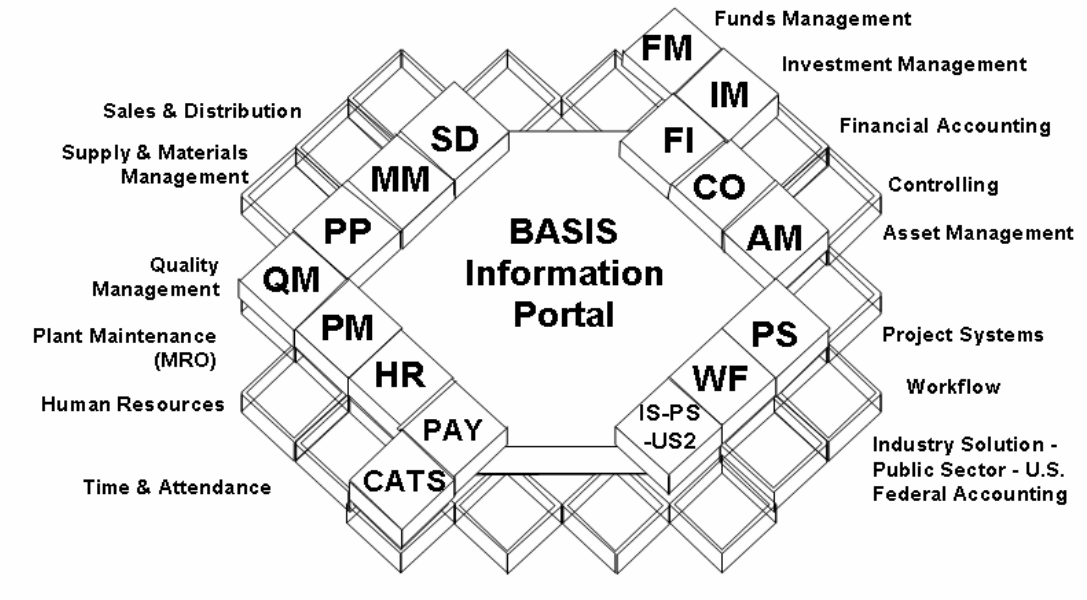

Figure 3. SAP Solution (NEMAIS)

a. ABAP/4 Data Dictionary

This data dictionary contains tables, domain definitions, field specifications, screen formats and specifications of reports and other data objects, used in programs coded in the ABAP/4 programming language (ASAP World Consultancy, 1999, p. 68).

b. The BASIS System

The SAP BASIS System controls hardware and its applications running on the SAP System. It is responsible for the runtime environment and it provides the following services;

- System administration
- Database administration
- BASIS services and communications
- ABAP/4 Development Workbench
- Business Engineering Workbench (ASAP World Consultancy, 1999, p. 70)

The BASIS System provides SAP the capability to control various hardware configurations.

c. Components of the Main SAP R/3 Applications

SAP provides numerous business applications that can be connected to the BASIS Information Portal. Each application is in the form of modules and is designed for specific functions. Below are the most common modules and a brief description of their function:

- Sales and Distribution (SD) provides the functionality for order completion, tracking, delivery and billing.
- Material Management (MM) facilitates procurement and inventory functionality used in daily business transactions.
- Production Planning (PP) provides functionality to plan and manage both inventory and production.

- Plant Maintenance (PM) provides functionality to plan and process maintenance actions.
- Financial Accounting (FI) provides functionality needed to satisfy legal requirements for the publishing of financial documents.
- Controlling (CO) provides functionality needed to control costs and revenue in the business.
- Investment Management (IM) provides the ability to manage tangible fixed assets and or financial investments.
- Project System (PS) this is ideal for project research or project development, modules in this application include, Basic Data, Operational Structures, Project Planning, Approval, Project Execution/Integration and Information Systems.
- Human Resources (HR) provides a complete management system for human resources.
- Quality Management (QM) provides quality control processes in modules that include Planning tools, Inspection Processing, Quality Control, Quality Certificates, and Quality Notifications.
- Industry Solutions (IS) provides components that apply to the industry for which SAP applications will employ, such industry may include Public Sector, Hospitals, Banks or Real Estate Management (ASAP World Consultancy, 1999, pp. 72-75).

Applications and modules employed depend on the type and scope of the SAP System being implemented. The combination of applications and modules can provide almost any desirable integrated standard business software.

2. SAP Basic Principles

According to SAP, their solution includes functions to support a full range of requirements from users. These functions can range from controlling user interfaces and dialog, to maintaining an integrated database system, and

providing “higher order statistical and control functions” (ASAP World Consultancy, 1999, p. 34).

SAP also notes that SAP R/3’s efficiency is the result of its basic design that includes multi-tier client/server architecture, open system principles, portability across operating systems, databases, presentation front ends, integration with distributed applications, and its ability in uncoupling applications, the front end, and databases.

a. Multi-Tier Client/Server Architecture

SAP R/3 relies on client/server architecture applied at different levels. Inherently such architecture is modular. In SAP, this architecture and its modularity are achieved through software so “the modes of interaction between the various clients and servers can be controlled” (ASAP World Consultancy, 1999, p. 34).

b. Open System Principles

In an effort to comply with international standards for open interface, the SAP R/3 System includes the following standards:

- Transfer Control Protocol/Internet Protocol (TCP/IP)
- Remote procedures calls (RPCs)
- Common Programming Interface-Communication (CPI-C)
- Structure Query Language (SQL)
- Object linking and embedding/dynamic data exchange (OLE/DDE)
- X.400/X.500, Messaging Application Programming Interface (MAPI), and Electronic Data Interchange (EDI)
- Other Open Interfaces needed for specialized applications, such as Computer-aided design (CAD), and Optical archiving (ASAP World Consultancy, 1999, p. 34)

Open system principles allow SAP the flexibility and portability to be interconnected to other SAP and non-SAP systems at the application, data or user interface levels. Such flexibility and portability comes with a price. Remote procedures calls allow transactions that should be blocked, to occur through firewalls. An example is when using XML and HTTP to conduct RPC calls using Simple Object Access Protocol (SOAP) (Hunter, Cagle, Dix, Kovack, Pinnock and Rafter, 2001, p. 27).

c. Portability Across Operating Systems

Per SAP documentation, the SAP R/3 System can be operated on UNIX, MPE/iX, Open VMS, OS/400, and Windows NT operating systems (ASAP World Consultancy, 1999, p. 35).

d. Portability Across Databases

SAP documentation states that the SAP R/3 System functions with IBM's DB2, Informix, Oracle, Software AG and Sybase database systems.

e. Portability Across Presentation Front Ends

According to technical documentation, the SAP R/3 System can use most presentation front ends, including Macintosh, OS/2PM, OSF/Motif and Windows, to display output.

f. Integration with Distributed Applications

SAP assures that their SAP R/3 System has the ability to coordinate with other SAP or Non-SAP systems to ensure that "both data and business functions are consistent" in a cluster by using Application Link Enabling (ALE) technology (ASAP World Consultancy, 1999, p. 36).

g. Uncoupling Applications, Front End and Databases

Per SAP, the SAP R/3 system has been designed recognizing the existence of a "diversity of hardware"

(ASAP World Consultancy, 1999, p. 36). Therefore, its functionality prevents “difficulty in uncoupling the application logic from the presentation system and the database configuration” (ASAP World Consultancy, 1999, p. 36).

In order to accomplish this, the SAP R/3 System may contain the following;

- Dedicated Database Servers
- Dedicated Application Logic Servers
- Special Task Servers
- Presentation Servers

The ability to uncouple applications, front ends and databases provides the flexibility to mix and match independently developed systems.

3. Provisions for Continuous Business Development

According to SAP documentation, SAP was designed with the capability to adapt to changing business needs. Supposedly, SAP can describe the current As-Is process and recommend modifications for future needs regardless of whether changes are driven by technology, people or processes.

SAP documentation also claims, that once future needs have been identified, SAP can be adapted accordingly by using Enterprise Data Modeling and other specialized functionality to adapt software.

a. Enterprise Data Models

According to documentation, SAP System contains an information model of itself that can be compared to the actual model of the company’s processes. After comparing, SAP can serve as a guide in the designing and implementing of new model to meet new requirements.

b. Tools for Adapting Software

SAP relies on the SAP R/3 Analyzer, SAP R/3 Reference Model and the IMG-Implementation Management Guide tools to assist in the customization of the software. SAP compares existing and target configurations, allowing users to modified parameters. The SAP documentation claims that this customization is accomplished without “altering any code” (ASAP World Consultancy, 1999, p. 39).

4. Reliability and Security

SAP believes that the SAP R/3 System will become the center of the company, and therefore, it is critical that it is designed to be reliable and available to all users when required. SAP claims that the integrity and confidentiality of data is paramount. Therefore, the SAP R/3 System is designed so that “no data or software code must be altered or corrupted, either intentionally or by accident” (ASAP World Consultancy, 1999, p. 60).

a. Functionality of the Software

According to SAP documentation, SAP ensures robustness in its software by the methods it uses to “design and build it” and by “extending the scope of the prerelease testing” (ASAP World Consultancy, 1999, p. 60).

b. Security Levels and Confidentiality

SAP documentation notes that the SAP R/3 System addresses security at the following levels:

- Desktop presentation system
- Application
- Database
- Operating System
- Network

SAP manages security at all these levels by the following security measures:

- R/3 internal security service, which concerns the desktop systems, application servers, database servers, and network communications at the application level.
- Database security services, which are provided by the database computer.
- System security services, which are assisted by the ease with which the R/3 system can be reconfigured without a loss of service if any subsystem comes offline.
- Network security services (ASAP World Consultancy, 1999, p. 60).

c. Reliability and Availability through Support

SAP R/3 includes a Computer Center Management System (CCMS) which is used to monitor, control and check the system including the database, operating system and network. SAP maintains a log of all transactions and it provides remote support via the Online Service System (OSS) which assists at the Application, Database, Operation System and Network level (ASAP World Consultancy, 1999, p. 61).

The SAP R/3 Solution, along with its basic design principles, flexibility, reliability and security, is being tested by the four ERP Pilot programs from SYSCOMS.

C. SMART ERP PILOT PROJECT

NAVSUP and NAVAIR teamed-up to modernized Aviation Supply and Maintenance by integrating aviation maintenance planning and supply support processes with the SMART ERP Application.

The SMART ERP Program replaced legacy applications that were based on 1960’s technology with a single integrated system that uses the SAP R/3 backbone. The

pilot program was limited to about 400 users and to Intermediate and Depot repairs of two systems: the E2C Aircraft and LM-2500 Gas Turbine Engine.

The scope of the project was divided into four phases. Phase 1 focused on selecting the software, phase 2 identified the core functionality, phase 2.5 determined any additional functionality and phase 3 deployed the program Navy-wide. Only Phase 1.0 and Phase 2.0 have been completed of the four phases. Phase 2.5 and Phase 3.0 have been put on hold awaiting the decision to converge with the other three ERP projects.

1. PHASE 1.0

Phase 1.0 was accomplished in less than one year. In addition to providing the Concept of Operations, and Business Case Analysis, it also determined software to be used and selected the areas of developmental opportunity.

The two key software components selected were SAP R/3 Version 4.6c to provide the backbone, and Manugistics, to provide the Advance Planning and Scheduling (APS) software.

a. SAP Software

SAP R/3 software was selected to provide the backbone to the ERP application. The SMART Project implemented the following six modules in the application:

- Sales and Distribution (SD)
- Material Management (MM)
- Production Planning (PP)
- Plant Maintenance (PM)
- Financial Accounting (FI)
- Controlling (CO)

The combination of the above six modules defines the foundation of SMART ERP and dictates conduct of daily

business transactions. To complement the SAP ERP System, Manugistics was chosen to provide the planning tool suite.

b. Bolt-Ons

The Manugistics APS modules provide the mathematical formulas needed to assist in the following decisions:

- Forecasting
- Demand Planning
- Budget Planning
- Supply Planning
- Transportation

APS functionality facilitates decision making by providing information that allows comparisons between options and their projected outcomes. Such comparisons may include determining the most efficient option between buying versus making, replacing versus repairing for a specific instance. The APS functionality specifically provided the SMART Project with the following:

- A tool set to improve the tracking of repairs, procurements and end of life cycle actions.
- Modeling capabilities based on resources constraints.
- Flexibility by allowing modeling and forecasting on different segments of the data.
- Ability to simulate various scenarios for comparison.
- Planning capability based on projected availability of resources or working toward specific goals set.
- Ability to demonstrate varying resources and their impact on mission performance.

Manugistics and its APS tool intended to provide the SMART ERP Project with “wholesale planning, order

fulfillment, and demand forecasting as well as [provided by] current legacy tools and…support business rules that represents the Navy environment" (NAVSUP, 2003, Slide 31 notes).

2. PHASE 2.0

In January 2003, the key milestone of "Going Live" was accomplished, effectively completing phase 2.0. The overall goal of this phase was to implement the pilot program for the E-2 aircraft and the LM-2500 gas turbine engine.

This phase entailed the design, configuration, testing and implementation of the pilot program. It relied on the Accelerated SAP (ASAP) guidelines for implementation. Such procedures include Project Prep, Blueprint, Realization, Final Prep and Go Live (McGrath, 2001).

ASAP is used to rapidly implement SAP R/3 System. It contains a project plan and checklist for the entire implementation. It uses a methodology different than what traditionally has been used. The plan begins by selecting SAP R/3 components that will meet business requirements. It does not entail conducting exhausting "As-Is" research to map out processes before implementation begins (ASAP World Consultancy, 1999, p. 650).

D. NEMAIS ERP PROJECT

NAVSEA set out to revolutionize Navy Ship Maintenance with the NEMAIS ERP Project. NAVSEA envisioned an efficient, reliable, centralized and secured global network supporting ship maintenance that would be accessible from anywhere at anytime.

The goal of the NEMAIS ERP System was "fleet-wide adoption and standardization of common processes and

procedures, rather then fleet-unique or depot-unique ways of doing business" (Keeter, 2002, p. 86) Business drivers for NEMAIS were:

- Lower budgets
- Mandated Savings
- Increased Workload
- Fewer ships and personnel
- Revolution on business affairs

The NEMAIS ERP Project was NAVSEA's response to CBPWG's objectives by:

- Providing Timely and Rapid Access to Information and Readiness Metrics.
- Supporting Total Asset Visibility.
- Enhancing the Planning and Scheduling Process.
- Providing Better Decision Making Tools.
- Reducing the Total Cost of Ownership.
- Minimizing and Simplifying Data Collection.
- Utilizing Common Processes Across the Enterprise.

The scope of NEMAIS involved integrating ship maintenance at all levels, including Intermediate, Depot and Overhaul activities, totaling over 28,000 users. NEMAIS also encompassed about 10,500 work centers afloat and over 99 legacy software systems.

NAVSEA, with guidance from IBM, selected the Method BLUE Methodology to implement the NEMAIS Project. Method BLUE is used to implement integrated projects that affect people, processes and technology.

Method Blue focuses on six areas for successful ERP implementation. Areas include Business, Organization, Application, IT Architecture, Engagement and Production.

The purpose of the Business Domain is to align a project with business needs. The Organization Domain analyzes organizational structure and manages changes needed to align the organization properly to project goals. The Application Domain is charged with providing a packaged solution that meets the business needs. The IT Architecture Domain is responsible for the IT infrastructure needed to support new applications. The Engagement Domain is responsible for project management and project success. The Production Domain focuses on providing a lasting application and its supporting environment.

The implementation of the NEMAIS ERP Project was to be conducted in six phases, Phase A through Phase F. The implementation phases represented regions or sites with similar functions as follows:

- Phase A: Mid Atlantic Regional Maintenance
- Phase B: Norfolk Naval Shipyard
- Phase C: Legacy Data Conversion (Concurrent with Phase B)
- Phase D: Remaining Maintenance Regions
- Phase E: Supervisor of Shipbuilding Sites
- Phase F: Afloat Enterprise Resource Planning (300 Navy Ships)

To date, a major part of the effort of the NEMAIS Project has been focused on Phase A, and although it is not completed, it has been the building block of this ERP project.

1. PHASE A

Phase A was focused on the Intermediate Maintenance Level activities in the Mid-Atlantic Region but it also emphasized the minimization of reconfiguring in later

phases (Navy Enterprise Team Ships, 2001). This emphasis provided the program flexibility to grow as opportunities arose. The overall intent of this phase was to build an "extendible solution in the Mid-Atlantic region that can be deployed Navy-wide, across ship and shore organizations" (Navy Enterprise Team Ships, 2001, p. 10).

NAVSEA also selected SAP R/3 as the backbone of the NEMAIS ERP System. The complete ERP solution also included the I2 Rhythm Optimal Scheduler to synchronize operation, an Oracle Database, and six bolt-on systems (Thomas, 2000).

a. SAP Software

Similar to SMART, NEMAIS was also dependent on the SAP R/3 Software for core functionality. NAVSEA selected the following modules to incorporate in the SAP R/3 application:

- Sales and Distribution (SD)
- Material Management (MM)
- Quality Management (QM)
- Plant Maintenance (PM)
- Human Resources (HR)
- Time and Attendance (CATS)
- Funds Management (FM)
- Investment Management (IM)
- Financial Accounting (FI)
- Controlling (CO)
- Asset Management (AM)
- Project System (PS)
- Workflow (WF)
- Industry Solution-Public Sector-US Federal Accounting (IS-PS-US2)

b. Bolt-Ons

The following bolt-on systems were added to supplement the SAP solution:

- Documentum used to manage documents.
- PlantWare used to monitor Hazardous Material.
- OROS 99 needed to enable Activity Based Costing.
- Abaco used to provide RF Barcode capability.
- MQ Series Integrator used to interface legacy systems.
- JetForms used to provide form management capability (Thomas, 2000, Slide 24).

These six additional applications provided a complete integrated solution that met the requirements of NAVSEA.

2. Beyond PHASE A

The future phases of NEMAIS are also awaiting the decision on ERP Convergence with the other three ERP pilots.

E. SIGMA ERP PROJECT

NAVAIR provides “program management, concept exploration, test and evaluation, and in-service support from concept development to disposal” (Carlton, 1999, p. 12) for airplanes, weapons and other systems to Naval operational forces.

NAVAIR responding to CVPWG tasking, and trying to improve itself in its efforts to continue quality service with dwindling resources, also selected an ERP pilot program with SAP core functionality. Furthermore, the ERP appeared to provide a solution to commonly recurring problems, which included:

- Disparate database.
- Suspect data integrity.

- No linkage between Financial, Maintenance and Supply Data.
- Multiple sources of data often resulting in multiple answers.
- Long lead time of results which were often conflicting.

The pilot program became know as the SIGMA ERP Project. It focused on evaluating ERP software in regards to program management processes and linking capabilities between contracting and financial systems. Specifically, NAVAIR hoped that the SIGMA ERP Project could provide the following:

- Ability for program managers to budget, plan, track execution, and measure performance across the organization.
- Ability to track configuration and assets across the Navy.
- Better cost visibility and more agile execution.
- Ability to track financial execution across the general fund and NWCF.
- Document tracking for milestone decision preparation.
- Fixed assets management.
- Ability for management to roll up financial performance and asset visibility.
- Ability to order MILSTRIP.
- Ability for planning work, capacity loading, and schedules.
- Support employee self-service.
- Reduce turn around time for time sheet adjustments.
- Verify that the [SAP proposed] three company code structure supports the team financial requirements.

NAVAIR, with the guidance of KPMG Consulting, one of the world's largest consulting companies, set the following ERP Implementation principles:

- No code modification to SAP software.
- There must be willingness to change business processes.
- Implement best practices.
- There must be acceptable justification for functionality.

NAVAIR sectioned the project into three tracks that included Project Management Track, People Management Track and Technology Management Track. The project implementation methodology entailed six steps.

- The first step was Project Preparation which defined objectives and success criteria, developed strategies and started the project.
- The second step was Business Blueprint which contained the project team training, preparing the system, process walkthrough and defined the scope of the project.
- The third step was Realization which included Design and Construct and Developing Interfaces.
- The fourth step was Final Preparation which involved Integration Testing and Preparing Production System.
- The fifth step was Go Live and Support which focused on End User Training and ensuring a productive system.
- The sixth step was Benefit Analysis which was design to analyze potential benefits (Erk, 2001).

The SIGMA ERP Pilot Project was to be rolled out in three phases. The first phase, the pilot phase, involved, approximately "7,000 users at 79 sites including sites in Japan and Italy" (Verton, 2002, p. 20). Phase two, Version 1.1/1.2, was to incorporate NAVAIR Warfare Center into the

solution. Phase three, Version 2.0, would bring the aviation depot community online (Caterinicchia, 2002).

1. SIGMA Pilot Version 1.0/1.1

In October 2002, the SIGMA Pilot Version 1.0 was rolled out. Like the other pilot projects, SAP R/3 Version 4.6c was selected as the core for the ERP system. SIGMA ERP also included an Oracle database system and various bolt-on systems.

a. SAP Software

NAVAIR selected the SAP R/3 application with the following modules in the integrated solution;

- Financial Accounting (FI)
- Funds Management (FM)
- Controlling (CO)
- Project Systems (PS)
- Material Management (MM)
- Sales and Distribution (SD)
- Human Resources (HR)

b. Bolt-Ons

The following bolt-on systems were added to supplement the SAP solution:

- ePower
- OROS use to enable Activity Based Costing
- MQ Series Integrator use to interface legacy systems
- JetForms use to provide form management capability

2. SIGMA Pilot Version 1.2

In January 2003, SIGMA Version 1.1/1.2 was deployed adding “15,000 users” from the Naval Warfare Centers (Verton, 2002, p. 20). Phase three has not been rolled out

and is awaiting decision on ERP Convergence with the other three ERP pilots.

F. CABRILLO ERP PROJECT

SPAWAR is responsible for a wide range of activities that involve the C4ISR (command, control, communications, computers, intelligence, surveillance, and reconnaissance) spectrum. Functions include research, design, development and life cycle support of systems (Oxendine and Hoffman, 2002).

Like the other SYSCOMS, SPAWAR was also elected to evaluate an ERP System. The SPAWAR ERP Pilot, known as Project CABRILLO, focused on financial management processes. The objectives of Project CABRILLO were as follows:

- Re-think and re-engineer processes applying best business practices.
- Use COTS software with no modifications.
- Establish common processes with end-to-end process integration and connectivity.
- Design for scalability, extensibility, and application at other Working Capital Fund activities.
- Single point of data entry and integration.
- Timely and accurate business information.
- Improved reporting and management tools (ABC, EVM).
- Reduce the number of business systems and interfaces.
- Eliminate manual processes.
- Provide automated workflow.
- Improve speed of processing business.

- Meet applicable federal financial management regulations, accounting standards, and requirements.
- Implement the U.S. Standard General Ledger (USSGL).
- 100% drill down capability to original transaction event: all transactions [must] have audit logging and trail.
- Utilize Joint Financial Management Improvement Program (JFMIP) certified software (Defense Finance Accounting Services (DFAS), 2003, Slides 10-13).

The Project CABRILLO Pilot implementation was called WAVE 1 and consisted of the following three phases.

- Phase one entailed conducting a business case analysis.
- Phase two included identifying the "As-Is" business process, demonstration of the software, and selection of the software integrator.
- Phase three dealt with ERP application procurement, configuration, and installation (Oxendine et al., 2002 and Defense Finance Accounting Services (DFAS), 2003).

1. WAVE 1

WAVE 1 was accomplished in June of 2001. Similar to the other three pilot ERP projects, the core of the ERP Application consisted of the SAP R/3 System. The system integrator chosen was Price Waterhouse Coopers.

The SAP ERP application included the following modules:

- Sales and Distribution (SD)
- Material Management (MM)
- Funds Management (FM)
- Human Resources (HR)
- Project System (PS)

- Workflow (WF)
- Investment Management (IM)
- Financial Accounting (FI)
- Controlling (CO)
- Fixed Asset Management (AM)

In addition to SAP ERP handling all of the daily transactions, WAVE 1 also resulted in all required external interfaces implemented and was made fully functional. By implementing the SAP integrated solution, SPAWAR "retired 34 systems, 20 instances of ORACLE databases, 37 interfaces and over 100 manual processes" (Defense Finance Accounting Services (DFAS), 2003, Slide 9).

2. Beyond WAVE 1

Like the other SYSCOMS, SPAWAR is also awaiting and working on the determination of interoperability issues amongst the four ERP pilots to be resolved before it proceeds with Project CABRILLO.

G. CONVERGENCE OF THE FOUR ERP PILOTS

In August of 2002, Assistant Secretary of the Navy (ASN) Research, Development and Acquisition (RDA) directed the convergence of the four Navy ERP Pilots. In September of 2002, the Navy Enterprise Convergence Team (NECT) was formed, and in December of 2002, the Chief of Naval Operations and Secretary of the Navy concurred and issued full support for the convergence decision.

NECT, who reports to ASN (RDA) through the Enterprise Resource Planning (ERP) Executive Steering Group (ESG), was tasked with the following missions:

- Develop a convergence plan for the Navy ERPs.
- Identify and document common business processes and unique business processes.

- Identify and document those areas where statutes or regulations preclude common processes.
- Coordinate Navy ERP architecture with other Navy and Departmental initiatives.
- Develop a Navy ERP acquisition strategy.
- Maximize reuse and integration of existing Navy related ERP documentation and resources.

Aside from being a directive, NECT emphasized the following reasons for the convergence of the ERP pilots:

- Optimize the Navy enterprise by focusing on the fleet, improved end-to-end product management, greater reengineering opportunities, and standardizing processes.
- Improve interoperability.
- Reduce total costs.
- Consistent approach to other initiatives.

The NECT convergence strategy is in its infant stages. It entails taking today's existing four ERP Pilots, normalizing them by providing a common solution by using Global Accelerated SAP Approach. In July of 2003, the Navy was still searching for a solution on how to integrate the four systems together (French, 2003).

Chapter IV will explore the Secure Application Integration Methodology (SAIM) to determine if it could be applied to the Navy's ERP integration efforts.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. APPLYING SECURE APPLICATION INTEGRATION METHODOLOGY TO DON EAI EFFORTS

A. INTRODUCTION

Chapter III discussed the Navy's current initiative to integrate the Enterprise Resource Planning (ERP) pilot project implemented by the four SYSTEMS COMMANDS (SYSCOMS). This chapter will discuss Secure Application Integration Methodology (SAIM) as a possible guideline to be used by the Navy Enterprise Convergence Team (NECT) in its efforts to achieve ERP interoperability.

B. SAIM PRINCIPLES AND NAVY'S IT GOALS

SAIM's guiding principles include align IT with the enterprise business strategy, build a solid enterprise architecture, leverage legacy and commercial software, while focusing on security. These four principles are similar to what an integrated ERP solution in the Navy should entail.

The Department of Navy (DON) Information Technology Standards Guidance (ITSG) established the principles of the Navy's ERP integration. DON ITSG is responsible for compliance with Defense Joint Technical Architecture (JTA) "which provides DoD systems with the basis for the needed seamless interoperability" (Percivall, 2002, p. 17).

In order to determine if SAIM and the Navy are driven by the same principles in regards to Enterprise Application Integration (EAI), the following four sections provide a comparison between the principles of SAIM and the requirements of DON ITSG.

1. Align IT with the Enterprise Business Strategy

Extensive efforts are being conducted throughout DOD and DON to get this right! In DON's Information Technology Standard Guidance (ITSG), this principle is specifically emphasized as follows;

> Every DON IM/IT process must directly link to and support the established DON mission, goals, and objectives. Information processes that are tied to strategic drivers-link to critical missions and extensive resources-will receive greatest emphasis for re-invention" (DON CIO ITSG Integrated Product Team, 1999, p. 6).

It has been noted that some organizations have implemented technology without considering its effects on the business (Boyle, 2003). SAIM provides explicit measures to ensure that the IT strategy is responsive to business goals.

2. Build on a Solid Enterprise

SAIM approaches enterprise architecture by focusing on building a solid infrastructure consisting of business components. It advocates an infrastructure that uses reusable business components to capture core functionality of the enterprise's business processes.

This SAIM principle is in line to support DON in establishing an IT architecture, which is needed to accomplish the ITSG task of supporting the Joint Tactical Architecture (JTA) by providing;

> The foundation for integration, interconnection, and interoperability between and among all tactical, strategic, and sustaining base systems, in-garrison and deployed, ashore and afloat; that produce, use or exchange information electronically (DON CIO ITSG Integrated Product Team, 1999, p. 3).

DON is relying on the architecture to provide the foundation for supporting the processes, systems and infrastructure needed for seamless connectivity within and external to the ERP pilot systems. SAIM provides specific guidance to assist in designing an enterprise IT architecture with complimentary business components.

3. Leverage Legacy and Commercial Software

NECT is mandated to implement best business practices, in a secure and manageable system that is economically sound and can interoperate with other systems. NECT must therefore strike a balance between current legacy applications that support a secure and manageable system, and COTS applications that provide economic efficiency and facilitate interoperability (DON CIO ITSG Integrated Product Team, 1999, p. 18).

Such a balance requires a proper match between legacy applications and COTS systems. SAIM provides guidance that facilitates analysis of applications and their ability to fit in the “To-be” design. This results in rapid alignment of outdated applications with modern ones.

4. Focus on Security

Security is paramount to both SAIM and DON. The DON ITSG requirement is as follows;

> In general, DON information systems should provide appropriate safeguards to ensure the confidentiality, integrity, availability, authenticity, and non-repudiation of information processed. The actual safeguards used should be commensurate with the operational requirements, information sensitivity level, and consequences of exploitation of the specific DON information system (DON CIO ITSG Integrated Product Team, 1999, p. 39).

SAIM addresses security in the context of cost, potential risks and business goals by applying "comprehensive and cohesive security analysis, risk mitigation, and integration techniques and tools" (Ruh et al., 2001, p. 157) throughout the project.

C. GOALS NOT COVERED BY SAIM

Four principles of SAIM support the Navy's goals for ERP convergence. However, DON ITSG specifically lists two other goals that are not in the principles of SAIM. These two goals are as follows:

> To promulgate standards and guidelines for system development and acquisition to significantly reduce lifecycle cost, shorten development and fielding time, and optimize the impact on program financial and execution performance (DON CIO ITSG Integrated Product Team, 1999, p. 3).
>
> To introduce guidance for measuring the effectiveness and efficiency of the Naval enterprise information infrastructure (DON CIO ITSG Integrated Product Team, 1999, p. 3).

In the following sections, the SAIM activity areas that support the principles will be evaluated at a high level. Evaluation will determine in what areas SAIM could be beneficial to NECT in the ERP convergence effort.

D. SAIM ACTIVITY AREAS

In support of its principles, SAIM focuses on the following five areas; Enterprise IT Strategy, Enterprise Architecture, Application Architecture, Component Development, and Application Integration and Deployment. Although all of these areas are equally important, the first two determine the foundation, and therefore, the quality of the last three.

This is particularly true in DOD and DON where incremental upgrades and improvements to IT have been the norm. Such piecemeal approaches can partly be blamed for DOD's commitment, or lack of, to past investments for reasons such as being locked into contracts or systems being too expensive to replace. However, such approaches also resulted from the absence of a DOD Enterprise IT Strategy and a DOD Enterprise Architecture.

1. Enterprise IT Strategy

SAIM, in this activity area, could assist NECT develop an IT Strategy aligned with Strategic Objectives. Once such an alignment is in place, SAIM also provides guidance to ensure the organization is ready to conduct effective EAI.

a. Identifying Strategic IT Initiatives

Strategic IT initiatives refer to "initiatives that clearly and directly support the overall business goals and strategies of the enterprise" (Ruh et al., 2001, p. 158). In DON ERP convergence efforts, this will not be an easy task for various reasons.

One of the reasons is the large number of senior stakeholders with varying interests. Another reason is the size and diversity of the SYSCOMS. Perhaps the most difficult obstacle to overcome is the establishment of standards that will provide interoperability amongst the traditionally autonomous SYSCOMS.

Establishing standards amongst the SYSCOMS will be difficult because it will entail addressing concerns of a very diverse and large group of stakeholders that includes civilian and military leaders, users, customers, technicians, operators and others, with very different

backgrounds. The concerns of stakeholders will have to be molded into a common understandable solution that can be used to design the system.

SAIM does not provide solutions for such “change management issues.” However, it does present some questions that need to be considered. Although the questions are geared toward a commercial organization, if questions are applied in the context of the Navy’s mission, they can be thought provoking and can serve as a basis for further discussion. Below are some questions that SAIM presents for attention.

- What are the enterprise’s competitive requirements?
- What is the basis of competition in the enterprise’s market space?
- How is the enterprise positioning the business within that environment?
- How does the enterprise deliver value to its customers?
- What is most important to each of the enterprise’s customer segments?
- What are the specific business goals and strategies of the enterprise?
- How does each goal and strategy address the competitive requirements? (Ruh, et al., 2001, p. 159)

The above questions raise three key issues regarding the Navy’s ERP convergence effort. The issues involve defining the enterprise, market space and business environment, and the competitive requirement to which the converged ERP will respond.

Defining the enterprise that will govern the converged ERP is critical to the employment and effectiveness of SAIM. NECT was tasked with the development of “a convergence plan for the Navy ERPs” and was given the responsibility to “coordinate Navy ERP architecture with other Navy and Departmental initiatives” (Rosenthal, Frye, Fitzpatrick, and Petz, 2002, Slide 28). SAIM requires a clear definition of the enterprise, which as per discussion with Kevin Fitzpatrick (personal communication, July 22, 2003) there needs to be more research to clear up.

Market space and the business environment also pose special challenges to the NECT that are not addressed by SAIM. The market and the environment to which the integrated ERP will respond are very diverse and dynamic.

The diversity is driven by geographical location and localized functional capacity. The dynamic behavior is driven by the unpredictability of a wide range of challenges that affect military resources.

Although SAIM promises “shorter development times of EAI-based applications” (Ruh et al., 2001, p. 160) to commercial organizations, the diversity and dynamics of the DON environment may still be too challenging.

In order to prioritize IT initiatives, SAIM compares requirements to strategy. Requirements that support the strategy are then analyzed for technology needs. This results in IT initiatives, which are then prioritized. The result is prioritized IT initiatives that support the strategy and the application integration.

In the Navy's environment, prioritizing such requirements will be difficult. In regards to NECT and its charter, the Secretary of the Navy's (SECNAV) areas of emphasis are people, combat capability, advanced technology and business practices (Rosenthal et al., 2002, Slide 4). Given the unpredictability of operational requirements and the broad spectrum of mission requirements, priorities often change. SAIM does not support such a change.

b. Assessing Readiness for EAI

SAIM provides a formal method to determine the enterprise's readiness to conduct EAI. This method focuses on three areas; business, organizational and technical issues (Ruh et al., 2001, p. 161). The Appendix provides SAIM EAI Assessment Criteria as a checklist that highlights important points for consideration prior to undertaking an EAI.

The Appendix is a good checklist that will generate difficult questions. Although the Appendix does not offer solutions or recommendations to NECT, it will assist to ensure that basic business, organization and technical issues are, at a minimum, brought forth.

2. Enterprise Architecture

SAIM's second activity area addresses the Enterprise Architecture. SAIM assumes that the application integration is being conducted in a homogenous enterprise comprised of components that can be brought together by a set of standards and systems interfacing.

It also assumes that an existing enterprise architecture is already in place and that all it requires is proper management “to constrain the designs of constituent components in order to achieve overall architectural goals” (Ruh et al., 2001, p. 162).

SAIM cannot assist NECT to design an Enterprise Architecture. Currently, integration of the ERP systems is being planned without having a definition of the enterprise or architecture, only guidance appears to be the compliance with the Joint Tactical Architecture (JTA).

Louise Reeder (personal communication, November 14, 2003) notified me that there are high level meetings where discussion involving ERP from other DOD agencies are being conducted, which would indicate that DOD would be the enterprise, and therefore, any architecture framework should be designed from a DOD point of view.

Regardless of how the enterprise is defined, SAIM can help NECT in developing a security policy, analyzing business component requirements, analyzing infrastructure requirements, assessing legacy and packaged applications and providing specifics to build the enterprise’s IT Architecture.

a. Developing a Security Policy

The security policy at the enterprise level must safeguard information assets to ensure achievement of business goals, regulatory compliance and mitigate risks to an acceptable level. SAIM recommends the following steps to create a security policy.

- Identify legal, regulatory, and business protection requirements that apply to the enterprise.

- Define roles and responsibilities.
- Identify critical information resources and place in categories based on their sensitivity.
- Identify protection goals for critical information resources.
- Determine the applicability of the policy.
- Define compliance requirements, describe acceptable and unacceptable use of information, and determine the consequences of unacceptable use (Ruh et al., 2001, p. 165).

Although the above are good guidelines, they do not address a major issue of concern to NECT, which involves the security risk caused by the aggregation of the data that occurs when the four ERPs are interconnected and provides the "big picture" as mentioned by Louise Reeder (personal communication, November 14, 2003).

This is a serious concern because unclassified tactical and operational data would be stored in a single database. In some cases, the single database and its metadata would provide sufficient details that would violate operational security.

b. Analyzing Business Component Requirements

SAIM advocates grouping "high-level business components" into clusters according to functions. Such functions should represent the fundamental business of the enterprise, such as Research and Development, Maintenance, Supply Distribution, and Finance and Accounting (Ruh et al., 2001, p. 166). This approach is very similar to the "Segment Approach" that is mandated by the Federal Enterprise Architecture Framework (Chief Information Officers Council, 1999, p. 4).

In regards to the Navy's ERP Integration efforts, in this area, SAIM falls short since it does not offer any guidance that facilitates standardizing components to support functions across the SYSCOMS. An example of such standardization would mandate Navy maintenance processes to be conducted the same by all SYSCOMS. NECT could benefit by specific guidance. The SYSCOMS, throughout all their organizational levels, have a tradition of emphasizing uniqueness and resistance to change.

Defining how the integrated ERP solution of the four SYSCOMS will fit into the enterprise, regardless of how the enterprise is defined, based on "clusters" and "segments" will prove difficult without proper motivation. SAIM does not provide any suggestions on how to motivate SYSCOMS. An example of motivation could be whichever SYSCOM performs a function best, obtains the funds and ownership of the applicable cluster or segment.

c. Analyzing Infrastructure Requirements

In this area, SAIM's goal is to "identify the characteristics that the enterprise must have in order to provide acceptable level of services" (Ruh et al., 2001, p. 166). SAIM recommends analyzing applications with respect to the following:

- Security requirements
- Maintainability and adaptability
- Reliability and availability
- Performance
- Scalability
- Integrity
- Manageability
- Usability

- Recoverability

Although SAIM provides a comprehensive list, according to Dr. Rick Hayes-Roth, there are other requirements that applications must support. These are Interoperability, Composability, and Functional Integration. In his paper, "Architecture, Interoperability, and Information Superiority," Dr. Rick Hayes-Roth, discussing driving technical requirements, provides the following;

> Interoperability-the architecture must facilitate the interoperation between constituent systems, including both those created in the future as well as those already deployed by the US and potential coalition partners.
>
> Composability-the architecture must facilitate the rapid creation of new applications and new processes in response to new missions and threats by allowing users to quickly compose off-the-shelf components in new ways, and easily modify and reconfigure systems and applications to meet changing missions and threats.
>
> Functional Integration-the architecture must support the integration of a large number of applications based on common business process, achieving complete coverage of the process while minimizing duplicated effort and resources (Hayes-Roth, 2003, p. 3).

Dr. Hayes-Roth's discussion directly relates to the Joint Technical Architecture (JTA) sponsored by DOD and applicable to NECT efforts. Therefore, SAIM would not suffice.

d. Assessing Legacy and Packaged Applications

The effort in this section focuses on ensuring that applications "fit to the overall design" of the enterprise architecture (Ruh et al., 2001, p. 167). It addresses two possible scenarios that may be encountered.

(1) Legacy systems completely subsume the business component. In this case, legacy systems are adequate or can be modified to perform a business component (e.g., Accounting Function). It also facilitates the ability for interfacing with a new application and/or components.

(2) Legacy systems partially supports business component. In this case, the legacy systems only support some of the business components and would require an interface to new business components.

SAIM also addresses issues with packaged applications, like ERP. It recommends the following questions for consideration.

- Does the legacy application provide an API?
- Is the API accessible via standard programming languages or via the vendor's proprietary language?
- Does the application support messaging, file-based input/output, or a database?
- Under what circumstances will the vendor support customer access to database tables?
- Is the vendor willing to commit to keeping the table definitions stable?
- If the design is multitier, are the internal interfaces well documented and supported?
- Does the application implement its own security scheme? If so, how difficult will it be to replace the existing security code with calls on standard authentication, authorization, and auditing services? (Ruh et al., 2001, p. 168)

NECT could benefit from the above checklist. NECT can also benefit from the assessment that each of the SYSCOMS accomplished in their way to implementing the ERP pilot project. Each SYSCOM assessed its own legacy and

packaged applications. However, NECT will receive no guidance from SAIM where it really needs it in the area of mapping applications to undefined business components.

e. Specifying the Enterprise IT Architecture

In this section, SAIM could provide the NECT a very useful and focused checklist. This checklist is useful for designing an IT Architecture that meets the standards provided by Joint Technical Architecture (JTA) and Federal Enterprise Architecture (FEA) directives.

The JTA and FEA provide very general top-level standards that if augmented with SAIM, could be used in designing a specific Enterprise IT Architecture. SAIM recommends the following views in organizing the IT architecture:

- **Component view.** Decomposes the architecture into components, which are then further decomposed into subcomponents, and so on.
- **Entity/object view.** Describes the business entities that are made available through reusable components in the business domain layer of the architecture.
- **User view.** Describes the user interface services that are available.
- **Data view.** Describes the database maintained in the architecture.
- **Legacy view.** Describes how legacy (and packaged) applications are integrated into the architecture.
- **Security view.** Describes the security services that are available, and provides guidance for how they should be used in applications.
- **Physical view.** Describes how the functionality and data are mapped onto physical devices.
- **System management view.** Describes how the various elements of the architecture are managed.

- **Disaster recovery view.** Describes how the architecture is transformed in the event of possible disasters, in order to maintain essential business capabilities.
- **Development view.** Describes how applications and components are developed (Ruh et al., 2001, p. 169).

JTA and FEA express general guidance concerning an enterprise IT Architecture in DOD. SAIM specifies views to be considered when designing the IT Architecture. NECT can benefit by using SAIM to identify specific IT Architecture objectives that will comply with JTA and FEA.

3. Application Architecture

An application in SAIM is "a software entity that focuses on providing a cohesive set of capabilities to end users" (Ruh et al., 2001, p. 170). According to Ruh et al., an example of such an application is "a line-of-business system (e.g., casualty insurance) or a front-end system that supports multiple back-end systems (e.g., customer service system)" (Ruh et al., 2001, p. 170).

In order to ensure a satisfactory Application Architecture, SAIM, for each application, conducts four tasks. The tasks include developing application requirements, analyzing application security requirements, developing the application architecture, and selecting commercial products. The following sections provide details on each of these tasks (Ruh et al., 2001, p. 170).

a. Developing Application Requirements

SAIM recommends using models of business processes and use case methods to determine application requirements. Ruh et al. state that "a use case is a description of a scenario in which an application will be use" (Ruh et al., 2001, p. 170).

The modeling of business processes and use cases entails more than capturing the current processes. Model and cases must also include information from the “To-Be” application. Although good advice, the NECT still needs more than just guidance. NECT needs the ability to capture business processes and a solid “To-Be” solution which with to work.

Although not provided by SAIM, there is a possible solution for identifying current business processes. Gulledge et al. used functionality in SAP to analyze interoperability issues between the SMART and SIGMA projects and offers a solution to capture the current business processes in the SAP solution (Gulledge, Simon, and Sommer, 2002).

The “To-Be” integrated solution is being defined. Until a clearer picture of what the “To-Be” solution will be, SAIM will have limited use in determining application requirements.

b. Analyzing Application Security Requirements

In this section, SAIM addresses threats and risks associated with proposed applications. Although not in detail, SAIM does emphasize some areas that can leave the integrated enterprise application vulnerable. Such emphasizes would benefit the NECT. Two of the areas addressed are the “weakness resulting from combining components” and “using proprietary security” (Ruh et al., 2001, p. 171).

c. Developing the Application Architecture

In this section, SAIM emphasizes the importance of associating, at a high level, components with applications. Ruh et al. define a component as “a software

entity that provides a cohesive set of functional capabilities through a specified interface" (Ruh et al., 2001, p. 141). According to Ruh et al., an example of a component would be the financial system of an enterprise.

SAIM does nothing more then serve as a reminder to ensure that the design of application-specific and reusable components is not taken lightly. It is important to remember that components may be used by other applications in the future. Special attention needs to be given to transactional protocols and the needs of existing applications.

In regards to this section, NECT could facilitate the integration by using the general concerns of SAIM to provide and enforce standards for the application architecture.

d. Selecting Commercial Products

SAIM provides little guidance in the selection of commercial products necessary to accomplish the application integration. It provides basic steps to follow, which include comparing and testing products against selected criteria. It does not provide any guidance on the type of tests or criteria that should be developed to validate products.

4. Component Development

As mentioned above, a component provides certain functional capabilities across the enterprise. SAIM has identified five different types of components and recommends a specific development strategy for each. Following are the five types of components and their development strategy as defined by SAIM:

- **Custom components.** These components are developed from scratch, at least initially. Since components can be reused across applications, any given component in an application development project may be built from scratch, be it a modification of an existing component, or to be reused without change.
- **Wrapped legacy application components.** These components are built on top of in-house legacy applications. The legacy may or may not have to be modified. As with custom components, the wrapper component may be new in a given application development or a (possibly modified) existing wrapper.
- **Wrapped packaged applications.** These components are similar in most respects to wrapped legacy application, but it is usually infeasible to modify the underlying legacy.
- **Wrapped databases.** These components are wrapped databases which are a special category that serves database information through a distributed object, messaging, or transactional interface. The wrapper is intended to relieve the client programmer of the burden of knowing the details of the database structure or how to access the database directly.
- **Infrastructure components.** These are usually off-the-shelf components such as ORBs [Object Request Brokers] and databases (Ruh et al., 2001, p. 174).

The important point that SAIM makes is that in designing components, long-term as well as current requirements need to be address. NECT could benefit from SAIM's general guidance, but will still need more specific solutions to design the components needed for the convergence of the ERP projects properly.

5. Application Integration and Deployment

In this, the final activity area, SAIM recommends evaluating a pilot program and performing security penetration tests. The next two sections address guidance

for pilot projects and SAIM's recommendation for minimal security items to be tested.

a. Evaluating a Pilot

The Information Technology Management Reform Act, also known as the Clinger-Cohen Act of 1996, delineates specific requirements that must be followed regarding the conduct of pilot programs. The following are excerpts from applicable sections of the regulation dealing with multi-agency, multi-activity conduct of each program:

- Each pilot program conducted under the title shall be carried out in not more than two procuring activities in each of the executive agencies (Sec 5301a2).
- Any pilot program may be carried out under this title for the period, not in excess of five years (Sec 5301c1).
- Before a pilot program may be conducted under section 5301, the Administrator shall submit to Congress a detailed test plan for the program, including a detailed description of the procedures to be used and a list of any regulations that are to be waived (Section 5302b).

The requirements above appear not to offer NECT the option of conducting a pilot project to test solution. A successful pilot program would involve entities from different activities or agencies, since the converged ERP solution must be compatible with JTA and FEA.

The five-year time limit for pilot projects is another issue that needs to be considered. The four ERP pilots conducted by SYSCOMS are at their time limit, and therefore, can no longer exist as pilot projects.

Lastly, NECT may not be ready to propose a pilot project since it has not defined the "To-Be" solution, and

therefore, may not be able to provide the needed documentation to justify the project.

b. Performing Security Penetration Tests

According to Ruh et al., the purpose of security penetration testing is to "test an architecture, as a whole, by attempting to defeat its security features" (Ruh et al., 2001, p. 176). SAIM recommends the following as the minimum that should be tested for security weaknesses:

- All perimeter access points should be checked to ensure authentication and access controls function properly.
- All network communications paths should be tested for vulnerability to disruption, corruption, or eavesdropping.
- All data entry, for example, forms and fields, should be tested for the ability to withstand bad entries of attempted form changes.
- All security mechanisms should be tested to ensure they support the security service described in the security policy.
- Common flaws should be checked, such as failure to remove default login accounts or test accounts.
- Known bugs in vendor products should be checked to ensure vendor patches have been applied.
- Session hijacking should be attempted for any session management techniques used, such as cookies or URL-encoded session IDs (Ruh et al., 2001, p. 176).

The above areas would provide a good starting point for security testing which could assist NECT in meeting the security requirements.

E. RISK MANAGEMENT AND UNPRECEDENTED TECHNOLOGY

SAIM uses the term "unprecedented technology" to refer to any unforeseen development in technology. An organization is at risk if it is not prepared to

incorporate such technology, if it needs it in order to keep up in the market or to gain a competitive advantage (Ruh et al., 2001, p. 176).

Ruh et al. recommends that any new applications, components, services or guidelines be addressed when attempting to mitigate risk regarding future developments (p. 177). Specifically, the following situations require special management attention:

- Business applications with extremely tight or near-term time-to-market constraints that depend on unprecedented elements.
- Mission-critical applications that depend significantly on unprecedented technology. In general, experience should be gained with new technology on applications that are not deemed mission-critical.
- Any applications that depend entirely on unprecedented elements (i.e., there is no identified contingency plan).
- Elements that are unprecedented not only within the enterprise, but in the marketplace as whole (i.e., “bleeding edge” technologies).

SAIM proposes the use of a prototype to manage unprecedented technologies because it “contains costs, ensures schedules, and provide early notification of problems that need special attention” (Ruh et al., 2001, p. 177).

In this area, SAIM supports the NECT in two ways. One way is by promoting the use of prototypes to mitigate the risk of unprecedented technologies. The other way is by addressing the Navy’s requirements of using Open Systems Architecture and Modularity in software intensive systems as mandated by the DOD 5000 series guidance.

F. ORGANIZATIONAL CONSIDERATIONS

EAI affects, and at the same time depends on the enterprise. The applicability and enforcement of SAIM can only be done by an enterprise approach and a unified effort from all the participants. Specifically, the proper execution of SAIM relies on the enterprise architecture organization and information security leadership (Ruh et al., 2001, p. 177).

1. Enterprise Architecture Organization

SAIM argues that in order "to effectively apply EAI, IT managers must develop a unified approach to managing the enterprise's architecture" (Ruh et al., 2001, p. 178). According to SAIM, an enterprise that exhibits the following has an organization that can properly support an enterprise IT architecture model:

> An Enterprise Architect, who reports to the CIO.
>
> An Enterprise Architecture Steering Committee, which is responsible for setting architecture policy and making major architecture decisions. This committee should include the Enterprise Architect (although not necessarily as the chair) and have representation both from the IT organization (development, maintenance, and operations group) and from the business units that IT supports.
>
> A small Enterprise Architecture staff, who are responsible for maintaining the enterprise architecture specification, for analyzing future requirements and associated architectural changes, and for evaluating new technologies. The staff should not be a permanent elite; instead personnel from various parts of the IT organization should rotate through its positions. In addition, a significant portion of staff members' time should be spent working on applications development projects. Working on projects severs two functions: It helps project personnel understand and apply architectural

> rules and guidelines, and it gives the architecture staff a realistic sense of the impact of architecture standards on the application projects (Ruh et al., 2001, p. 178).

NECT has been directed to provide a solution that complies with JTA and FEA. In reality, these two architectures only provide a sense of direction of where DOD is going (Hayes-Roth, 2003), and does not provide specific guidance.

Although the Cohen Act of 1996 established CIO and other governing bodies, it did not define the enterprise architecture organization. Nevertheless, NECT needs to ensure that its efforts comply with proper architecture design.

It is up to NECT to ensure that it designs a solution that will fit within the undetermined DOD enterprise architecture. SAIM is useful in this area by focusing NECT on asking the enterprise's organization for the proper guidance.

2. Information Security Steering Committee

SAIM considers that the management of security issues must be at an enterprise level to ensure consistency and proper application. It recommends an Information Security Steering Committee "be responsible for defining the enterprise's information security policy and for ensuring consistency with the business goals" (Ruh et al., 2001, p. 178). Per Ruh et al., the committee should include "executive-level management" and it should also be responsible for the following:

- Reviewing the enterprise's security policy in light of emerging threats.

- Investigating any breaches of security that may occur within the enterprise, and adopting rules to ensure that they are not repeated.
- Reviewing security characteristics of critical applications, both at the stage of requirements definitions and before the applications, are placed in production.
- Sponsoring the development of security education and training programs for the enterprise (Ruh et al., 2001, p. 178).

NECT could benefit by using the above recommendations to guide its efforts from both, the point of view of an enterprise or as the convergence team leading the project.

G. CONCLUSION

NECT is charged with integrating four ERP systems. SAIM principles and activity areas provide some benefits to NECT. SAIM also provides benefits by bringing forth issues regarding the Navy's enterprise IT architecture, application integration and deployment, risk management and organization factors.

Chapter V discusses conclusions and recommendations in using SAIM in NECT ERP convergence efforts. It also provides areas for future research to assist the NECT efforts

V. CONCLUSIONS AND RECOMMENDATIONS

A. SUMMARY

In response to the Joint Vision 2010 and Revolution in Business Affairs (RBA), the Navy selected Enterprise Resource Planning (ERP) systems to achieve its objectives. It chartered the Systems Commands (SYSCOMS) to implement pilot programs in order to assess the applications.

The pilots have been completed and are now at a critical juncture where they must be converged into a single system. The convergence of such applications is new to the Department of Defense (DOD) and Department of Navy (DON).

Secure Application Integration Methodology (SAIM) was designed to facilitate Enterprise Application Integration (EAI) by providing a blueprint that can be used as a guide. SAIM can be applied to some areas of the EAI efforts, but in other areas, it does not provide the needed guidance to be suitable for use by DON.

B. RESEARCH QUESTION

1. Primary Research Question: Can Secure Application Integration Methodology (SAIM) be Applied to the SYSCOMS' ERP Convergence Effort?

SAIM can be applied to the SYSCOMS ERP convergence effort. However, it does not address the following four areas that need to be considered for a successful DON ERP convergence effort:

- SAIM does not support all of DON CIO requirements.
- SAIM does not provide change management guidance.
- SIAM does not account for the Navy's unique operational environment.

- SAIM presumes that EAI is being conducted in an organization that has an Enterprise Architecture in place, from which the IT architecture can be derived.

These four shortcomings are significant and can be detrimental to the success of the Navy's ERP convergence effort. The following explains the four shortcomings.

a. DON CIO Requirements Not Supported

SAIM does not address the DON CIO requirements of managing lifecycle costs. It also does not provide guidance on how to conduct the EAI so as to optimize financial and execution performance. As stated in Chapter IV section C, these are two of DON CIO's major requirements.

b. Change Management Guidance

DON ERP's convergence effort will entail a culture change. Such culture change will require change management support to implement new processes. As stated in Chapter II, effective EAI will most likely require a change in how business is currently done. Such a change will involve many stakeholders and users that will need to be managed through the changes. SAIM does not offer any guidance on how this could be done.

c. Uniqueness of Navy Environment

The operational environment, to which the converged enterprise application will respond, poses special challenges not covered by SAIM. One such challenge is the dynamic environment in which mission requirements often change. A change in mission requirements may result in changes to the strategic IT initiatives.

This change would be difficult for SAIM to handle. SAIM uses prioritized IT initiatives that support

the strategy as the foundation for application integration. A change in IT initiatives, driven by a change in mission, would change the foundation of the application integration, thus forcing a major change to the EAI strategy. A major change in EAI strategy could result in expensive rework. At the worst, if the change is ignored, the results could produce an EAI that does not support the Navy's operational requirements.

d. SAIM Depends on the Enterprise Architecture

SAIM requires an Enterprise Architecture for properly analyzing the requirements of security, business components, infrastructure, legacy and packaged applications, and to develop the IT architecture. Such Enterprise Architecture is missing in DON, and therefore, the full benefits of SAIM cannot support the ERP convergence effort.

As noted in Chapter IV, the Navy Enterprise Convergence Team (NECT) received guidance to comply with other DOD applications. NECT must also comply with the Joint Tactical Architecture (JTA) and the Federal Enterprise Architecture (FEA) but the ERP converged solution is not part of any specific Enterprise Architecture in the sense that SAIM requires.

SAIM envisions an Enterprise Architecture that controls all IT systems in the enterprise. Even if Enterprise Architecture existed in DON, this level of control would be difficult, if not impractical, to accomplish for several reasons. These reasons include:

- The number of systems in DON is too large.
- The geographical dispersion of Navy assets is too great.

- Given all the functions that are accomplished throughout DON, there may be too many functional requirements to be captured by a single Enterprise Architecture.

The above three obstacles will make defining an Enterprise Architecture much more difficult, yet in order to use SAIM effectively, such definitions must exist.

2. Secondary Research Question: Can Using Secure Application Integration Methodology (SAIM) Mitigate Risk in the NAVY's ERP Convergence Effort?

Although SAIM will not satisfy all the needs of the ERP convergence effort, it can mitigate risk. It has four elements that could lower the failure risk in the Navy's ERP convergence effort. It mitigates the risk for the following reasons:

- As stated in Chapter IV, it provides a view of the complete EAI process from start to finish.
- It demonstrates the importance of an Enterprise Architecture in an EAI endeavor.
- It provides useful checklists that could serve as reminders to the NECT.
- It addresses important considerations for an effective EAI.

The following shows how the above four points could assist NECT in its efforts to conduct an effective EAI.

a. Complete Walkthrough

SAIM's step-by-step procedures can provide NECT a top-level overview of the process from start to finish. This would give NECT the lead for further research in specific areas as it deems necessary in its effort to tailor a suitable EAI methodology. This would help reduce the risk of NECT missing key EAI procedural steps.

SAIM's step-by-step procedures could also be used as training materials for stakeholders and users in preparation for the ERP convergence. Its simplicity, in approach and depiction, makes it suitable for technical and non-technical personnel. SAIM, as training material, would assist in lowering change management risk by communicating the EAI process to all involved.

b. Importance of an Enterprise Architecture

Perhaps the greatest assistance provided to NECT by SAIM is its emphasis on the requirement of an Enterprise Architecture as the foundation for the IT architecture. The importance of the architecture is echoed by Clements, Kazman and Klein as follows;

> Architecture is first and foremost a key to achieving system understanding. [Architecture] As a vehicle for communication among stakeholders, it enables high-bandwidth, informed communication among developers, managers, customers, users, and others who otherwise would not have a shared language. [Architecture] As the manifestation of the earliest design decisions, it is the key to project organization and the expression of strategies put in place to design the system (often using the design vocabulary of architectural styles). [Architecture] As a reusable, transferable abstraction of a system, it enables understanding of future systems that share the same architecture. Once built, an architecture and the components that populate it can be one of an organization's key assets for many years (Clements, Kazman, Klein, 2002, p. 15).

Having an architecture to guide the effort of NECT would greatly reduce the risk of failure that will otherwise exist if the ERP convergence is conducted without any architectural framework.

c. Useful Checklists

SAIM also provides checklists that can be useful to NECT. These checklists are presented as questions or as steps for consideration, and are easy to follow. The checklists cover general areas and provide special emphasis to common pitfalls, along with highlighted areas that, if ignored, would increase failure risks.

d. Important Considerations

SAIM provides advice that goes beyond the technicalities of EAI. It offers key points to consider involving security and unprecedented technology. These points provide a view to what Bosch labels as quality requirements, as he explains;

> Operational quality requirements are qualities of the system in operation, e.g. performance, reliability, robustness and fault-tolerance. Unlike functional requirements, quality requirements can generally not be pinpointed to a particular part of the application but are a property of the application as a whole (Bosch, 2000, p. 27).

SAIM mitigates failure risk by directing NECT to consider the quality requirement of the converged solution as a whole, and specifically, in regards to security and unprecedented technologies.

C. CONCLUSIONS

SAIM could be used by NECT in its ERP convergence efforts. However, SAIM does not provide a complete solution for NECT to effectively converge the SYSCOMS ERP projects. SAIM's shortfalls include non-compliance with all requirements of the DON CIO, it also does not provide any change management guidance, it is not suited for the Navy's unique operational environment, and it assumes that an

Enterprise Architecture is in place, which currently is not true.

On the other hand, SAIM would help NECT mitigate risk by providing an easy to follow step-by-step walkthrough of the EAI process. It would also make NECT aware of the importance of an Enterprise Architecture to EAI efforts. SAIM also provides a checklist that would be useful to the NECT. Lastly, SAIM lists important special considerations that in the end would also help NECT mitigate risk.

D. RECOMMENDATIONS

NECT should use SAIM to familiarize itself with the demands of EAI. NECT should pay particular attention to the checklists and areas that SAIM emphasizes. NECT should also use SAIM as a training aid to assist in the management of the ERP convergence project.

However, NECT should also be aware of SAIM's shortcomings. Specifically, NECT, before moving ahead with the convergence project, should resolve the Enterprise Architecture issue. It should request guidance to determine what Enterprise Architecture to follow, realizing that without an architectural framework, the success of the EAI project is at risk.

E. AREAS FOR FURTHER RESEARCH

In the analyses of SAIM in light of DON's ERP convergence efforts, some areas for further research have surfaced. These areas apply to the NECT convergence effort, EAI, in general, Enterprise Architectures and quality requirements. The areas are as follows:

- Key to the NECT ERP convergence efforts is an architectural framework to guide the integration. Should the Enterprise Architecture be at the Department of Defense level or Department of the

Navy Level? Should it be a DOD Enterprise Architecture or a DON Enterprise Architecture?

- Given the diversity and complexity of the functional requirements of DOD and DON, should there be only one Enterprise Architecture or only one Enterprise Application, instead of multiple architectures and multiple applications?
- Given that quality requirements of the overall Enterprise System are not addressed by any individual system, what would be the best way to ensure that quality requirements of the overall enterprise are satisfactorily accomplished?

APPENDIX SAIM'S EAI ASSESSMENT CRITERIA

EAI ASSESSMENT CRITERIA

Management

- ☐ Is there a coherent IS strategy or master plan?
- ☐ Is there a plan to transition from stovepiped applications to an EAI framework?
- ☐ Do line-of-business and IS personnel collaborate on planning?
- ☐ Are projects synchronized to leverage common functions and data?
- ☐ Is the culture collaborative?
- ☐ Is development of reusable components, and reuse of these components rewarded.

Organizational

- ☐ Are roles, responsibilities, and relationships related to EAI clearly defined?
- ☐ Is responsibility to plan, coordinate, manage, and execute IS activities enterprise-wide allocated to a Program Management Office?
- ☐ Is responsibility for the EAI architecture clearly defined?
- ☐ Is there a smoothly functioning CM function in place?
- ☐ Is there a smoothly functioning QA function in place?

Methodology

- ☐ Is a repeatable life cycle methodology in place?
- ☐ Does the methodology address application integration specifically?
- ☐ Does the methodology provide for definition and control of an enterprise IS architecture?
- ☐ Does the methodology provide a means to integrate unprecedented technologies into the enterprise mainstream?
- ☐ Are metrics collected and analyzed?
- ☐ Are risks explicitly managed?

Infrastructure and Technical Expertise

Are EAI technologies currently in use?

- ☐ CORBA
- ☐ DCOM
- ☐ MQ Series
- ☐ Other messaging
- ☐ Tuxedo
- ☐ Other transaction monitors
- ☐ EJB
- ☐ Translation/Conversion Services
- ☐ Data integration (Extract/Translate/Load tools)
- ☐ Is an EAI training program in place?

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

Applegate, L. M., Austin, R. D., and McFarlan, F. W. *Corporate Information Strategy and Management*. 6th ed. McGraw-Hill Irwin, 2003.

ASAP World Consultancy. *Using SAP R/3*. 3rd ed. QUE, 1999.

Bardach, E. *A Practical Guide for Policy Analysis: The Eightfold Path to More Effective Problem Solving.* Chatham House Publishers of Seven Bridges Press, 2000.

Bosch, J. *Design & Use of Software Architectures: Adopting and evolving a product-line approach.* Addison-Wesley, 2000.

Boyle, R. D. "Unlocking ROI." *APICS*, v. 13, pp. 37-39. June 2003.

Carlton, J. "Naval Air Systems Command: One Team." *Naval Aviation News*, v. 81, pp. 12-13. January/February 1999.

Caterinicchia, D. "NMCI Earns Kudos for Easing Software Rollout." *Federal Computer Week*, v. 16, p. 16. 11 November 2002.

Chief Information Officers Council. "Federal Enterprise Architecture Framework." (September 1999). Retrieved 10 January 2004 from http://www.cio. gov/documents/fedarch1.pdf]

Clements, P., Kazman, R., and Klein, M. *Evaluating Software Architectures: Methods and Case Studies*. Addison-Wesley, 2002.

Cook, G. "Trends in Business Technology Management: Enterprise Resource Planning, Outsourcing, Application Service Providers." Information Systems Economics, Naval Postgraduate School Course IS-4031, PowerPoint Presentation, 2003.

Crabtree, R. "Value Stream Mapping." *APICS*, v. 13, p. 22. July/August 2003.

Cummins, F. A. *Enterprise Integration: An Architecture for Enterprise Application and Systems Integration*. Wiley Computer Publishing, 2002.

Davenport, T. H. "Putting the Enterprise into the Enterprise System." *Prentice Hall Custom Business Resources*, pp. 159-169. Compiled by MSC 3824 Managing the Information Resource College of Business Administration Northeastern University Professor Leslie D. B. Winter, 2002. Pearson Custom Publishing, 2002.

Defense Finance Accounting Services (DFAS). "DON ERP." (July 2003). Retrieved 20 October 2003 from http://www.nfmc.navy.mil/PDI/PDI2002/14ERP.ppt.

Devaraj, S. and Kohli R. *The IT Payoff: Measuring the Business Value of Information Technology Investments*. Prentice-Hall, 2002.

DON CIO ITSG Integrated Product Team. "Information Technology Standards Guidance." (5 April 1999). Retrieved 10 January 2004 from http://www.doncio.navy.mil/PolicyMatrix/Uploads/1010PDO15817.pdf.

Erk, M. "Presentation to: The Society of Logistics Engineers, Enterprise Resource Planning." 16 November 2001.

Frame, D. J. *Managing Projects in Organizations*, rev. ed. Jossey-Bass Publishers, 1995.

French, M. "BearingPoint to Get $30 Million Deal." (21 July 2003). Retrieved 15 August 2003 from http://www.fcw.com/fcw/articles/2003/0721/web-navy-07-21-03.asp.

Gay, L. R., and Airasian, P. *Educational Research: Competencies for Analysis and Application.* 6th ed. Prentice Hall, 1996.

Gillmann, M., Herter, J., Jung, C. G., Kaufmann, G., and Wolber, M. "Cooking the Web-ERP." (2002). Retrieved 15 August 2003 from http://www-cf.usc.edu/~arslan/cs586/25190602.pdf.

Gralla, P. *How the Internet Works*. 6th ed. QUE, 2002.

Gulledge, T. R., Simon, G. and Sommer, R. A. "Using ARIS to Manage SAP Interoperability." (2002). Retrieved 15 November 2003 from http://www.pac.gmu.edu/presentations/UsingARIStoManageSAPInteoperability.pdf.

Hayes-Roth, R. "Architecture, Interoperability, and Information Superiority," paper presented as part of Information Systems Strategy & Policy: How to Be and Effective CIO or IT Strategist, Naval Postgraduate School Course IS-4182. 20 January 2004.

Hunter, D., Cagle, K., Dix, C., Kovack, R., Pinnock, J., and Rafter, J. *Beginning XML*. 2nd ed. Wrox Press Ltd., 2001.

Information Technology Management Reform Act, 40 U.S.C. 1401. Retrieved 12 December 2003 from http://wwwoirm.nih.gov/itmra/itmra96.html.

Jakovljevic, P. J. "Essentials ERP." (26 December 2000). Retrieved 10 March 2003 from http://www.technologyevaluation.com/Research/ResearchHighlights/BusinessApplications/2000/12/research_notes/TU_BA_PJ_12_29_00_1.asp.

Keeter, H. "Renaissance at NAVSEA." *Sea Power*, v. 45, pp. 85-88. April 2002.

Linthicum, D. S. *Enterprise Application Integration*. Addison-Wesley, 2000.

Malaika, S., Nelin, C. J., Qu, R., Reinwald, B. and Wolfson, D. C. "DB2 and Web services." *IBM Systems Journal.* v. 41, pp. 666-685. 2002.

McGrath, F. Phase III Presolicitation Document. (2001). Retrieved 20 October 2003 from http://www.nor.fisc.navy.mil/contracts/announce/Phase%20III%20Pre%20Solicitation%20Documentation%2016%20May%2001v3%20.doc.

NAVSUP. "SMART/ERP Workforce Overview." (2003). Retrieved 16 December 2003 from http://www.navsup.navy.mil/npi/prod_serv/supply/erp/ppt/SMART_ERP_Workforce_Overview.ppt.

Navy Enterprise Team Ships. "Final Draft: Scope and Approach Document." Document was prepared for Navy Enterprise Maintenance Automated Information System (NEMAIS), Norfolk Virginia on 11 June 2001.

NDS Systems. "Web Services." (April 2003). Retrieved 12 September 2003 from http://www.ndsapps.com/XML_Products/Web_Services.

Oxendine, E., and Hoffman, D. M., IV. *Enterprise Resource Planning (ERP: A Case Study of Space and Naval Warfare Systems Center San Diego's Project Cabrillo*. Master's Thesis. Naval Postgraduate School, Monterey, California, September 2002.

Percivall, G. "Architecture Standards for Information Systems." (June 2002). Retrieved 17 January 2004 from http://www.gst.com/library/arch_standards_is.pdf.

Plossl, W. G. *Orlicky's Material Requirements Planning*, 2nd ed. McGraw-Hill, 1994.

Ramankutty, P. "Perfect Union." *APICS*, v. 13, pp. 37-40. April 2003.

Rosenthal, R., Frye, B., Fitzpatrick, K., and Petz, M. "Navy ERP Project Overview: Presentation to Association for Enterprise Integration Conference." 19 September 2002.

Ruh, W. A., Maginnis, F. X., and Brown, W. J. *Enterprise Application Integration*. Wiley Computer Publishing, 2001.

Senge, P. M. *The Fifth Discipline: The Art and Practice of the Learning Organization*. Currency Doubleday, 1990.

SoftSelect Systems. "2003 ERP Software Scorecard." *APICS*, v. 13, pp. 61-69. June 2003.

Thomas, C. "NEMAIS ERP Presentation to Commanding Officer of SIMA, Ingleside Texas." 13 November 2000.

Verton, D. "ERP, Security Rollouts Boost Troubled Navy Intranet Project." *Computerworld*, v. 36, p. 20. 18 November 2002.

Worthen, B. “Nestlé’s ERP Odyssey.” (15 May 2002). Retrieved 21 May 2003 from [http://www.cio.com/archive/051502/nestle.html.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. Professor Dan C. Boger
   Naval Postgraduate School
   Monterey, California

4. Glenn R. Cook
   Naval Postgraduate School
   Monterey, California

5. Professor Douglas E. Brinkley
   Naval Postgraduate School
   Monterey, California

6. LCDR Ramon O. Marin
   Naval Postgraduate School
   Monterey, California